RSR04E/08E 配置手册

2005-8

目录

前言

本手册简要提供配置RSR系列路由器JUNOS 7.3 的信息，提供了一些配置例子，由于水平有限，时间仓促，可能会出现错误，如需要详细了解RSR JUNOS的配置信息请参阅JUNOS documentation CD

相关文档
RSR JUNOS 文档

JUNOS Configuration Guides
Feature Guide
JUNOS-FIPS
MPLS Applications
Multicast
Network Interfaces and Class of Service
Network Management
Policy Framework
Routing and Routing Protocols
Services Interfaces
System Basics
VPNs

1. JUNOS 简介

Junos 使用 FreeBSD 内核，模块化的设计，提供卓越的故障恢复能力并确保能够简单地集成 IPv6、MPLS 等新功能。

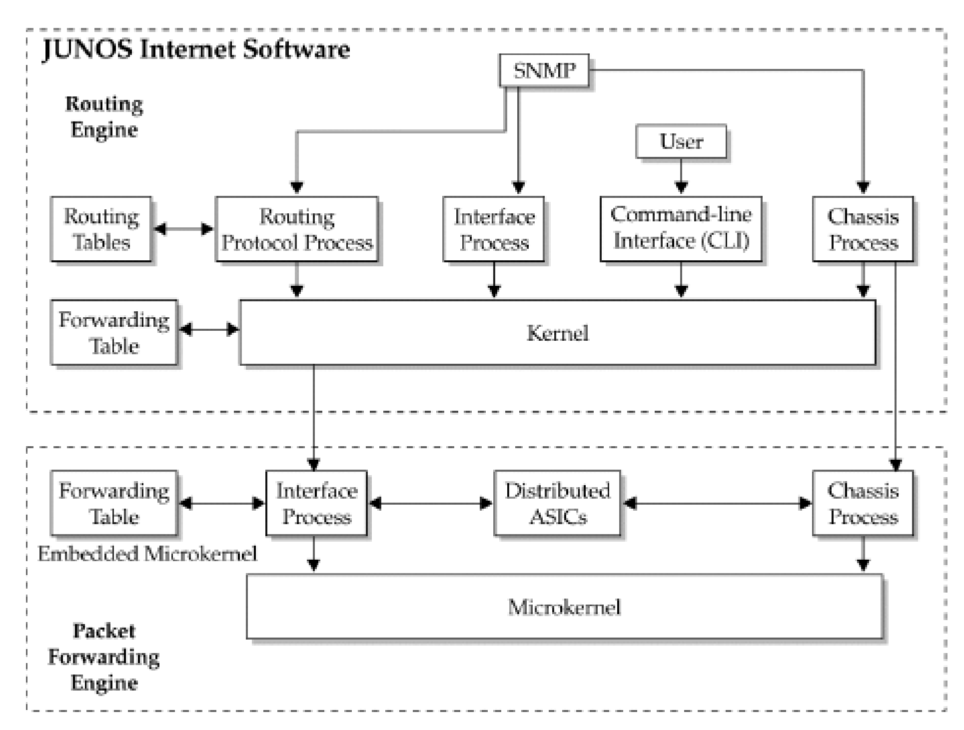


图 1-1

1.1 系统启动顺序

第一，移动介质，例如 PC-card，PCMCIA。
第二，Flash Disk。
第三， IDE 硬盘。

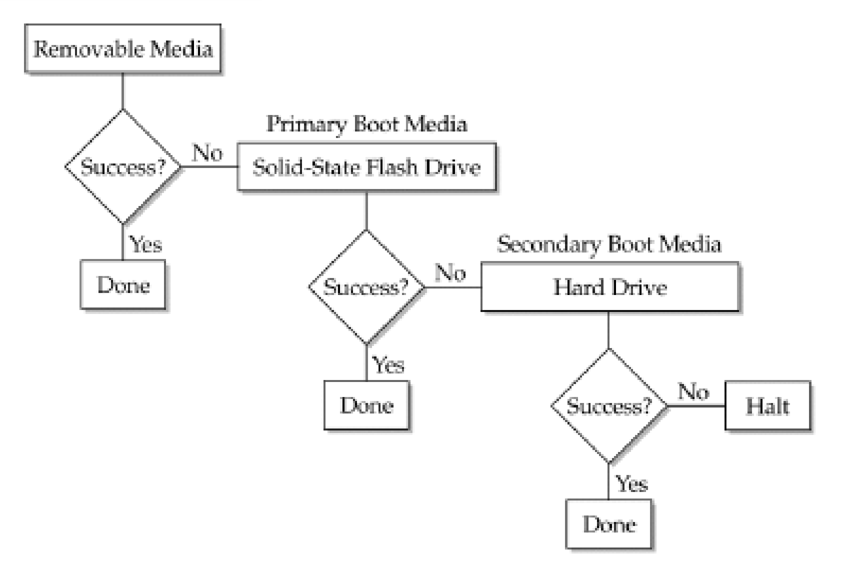


图 1-2

RSR 内部只有 IDE 硬盘， flash disk、pc-card 可以选配。默认从 IDE（笔记本硬盘）硬盘启动。

1.2 系统启动、关闭和升级

1.2.1 RSR 系统启动

加电。系统第一次启动只能通过 control 端口登陆。超级终端使用默认参数。

看到 login 提示符，输入用户名为 **root** 密码是空。进入系统后输入 cli 命令进入 RSR command line interface 可以进行设备调试。

login: **root**

Password:

Terminal type? [vt100] **y**

root%**cli**

root>

1.2.2，关机或者从启动。

使用的是 unix 内核需要通过命令操作完成关机或重新启动。如果系统掉电，可能设备会启动设备几率 1%。

(1) 系统关机

root>**request system halt**

Halt the system ? [yes,no] (no) **y**

root@1> request system halt

Halt the system ? [yes,no] (no) yes

*** FINAL System shutdown message from root ***
System going down IMMEDIATELY

Shutdown NOW!
[pid 4050]

The operating system has halted.
Please press any key to reboot.

1，出现 “The operating system has halted. ” 提示符，设备可以关电。
2，如果此时按任意键系统会重新启动。“Please press any key to reboot.”

(2)系统从启
  root>**request system reboot**
Reboot the system ? [yes,no] (no)**y**

1.2.3 更新系统软件

Junos 安装软件是 .tgz 的包集合。文件名类如：jinstall-7.1r1.4-domestic.taz
（1），升级首先通过 FTP 软件将升级文件上传到 RSR 目录就在/var/home/admin/下
其中 admin 是的你登陆名。
（2），使用 命令 **request system software add** <path-and-name-for-package>来升级软件。
从启动系统。
admin@rsr04e>**request system software add** jinstall-7.1r1.4-domestic.taz
installing package '/var/home/admin/ jinstall-7.1r1.4-domestic.taz'….
….
Restarting cli
进入 cli

1.3 使用 junos 帮助

admin@rsr04E> **help reference | topic**
example: admin@1> **help topic interfaces lo0**
Configuring the Loopback Interface

On the routing platform, you can configure one physical loopback
interface, lo0, and one or more addresses on the interface. To do this,
include the following statements at the [:

```
[edit interfaces]
lo0 {
    unit 0 {
        family inet {
            address loopback-address;
            address <loopback-address2>;
            ...
        }
    }
}
```

When   do not include a destination prefix.
Also, in most cases, do not specify a loopback address on any unit other than unit 0.

```
+-----------------------------------------------------------------------+
|         | NOTE: For Layer 3 vyou can          |
|         | configure multiple logical units for the loopback interface.       |
---(more)---
```

2. JUNOS CLI

junos 是通过 MGD 进程来处理用户发出的 cli 命令。

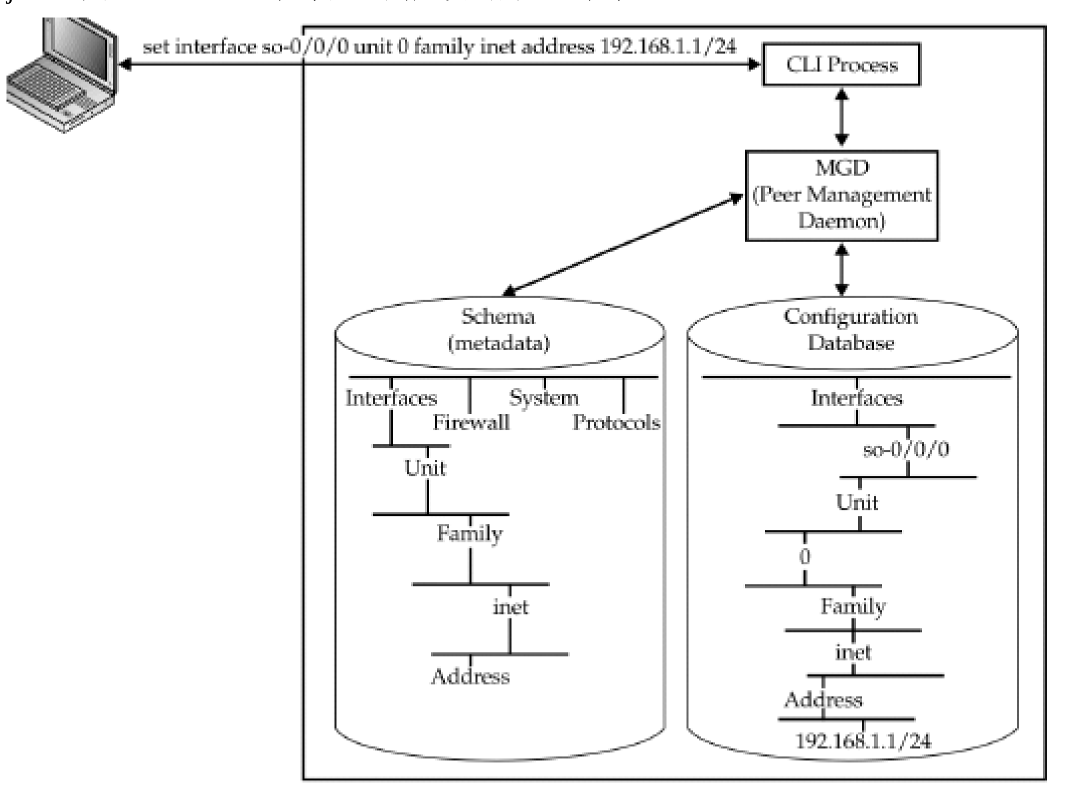


图 2-1

Example 配置一个接口：
set interfaces fe-0/0/0 unit 0 family inet address 192.168.1.1/24
delete interfaces fe-0/0/0 unit 0 family inet address 192.168.1.1/24

2.1 CLI 的模式

1，操作模式

监控和排错 软件，网络连接、路由器硬件。
2，配置模式
test@lab2>

配置路由器包括：interface、路由信息、路由协议、用户访问、系统硬件参数。
test@lab2> configure
[edit]
test@lab2#

2.1.1 CLI 操作模式

（1）命令层次

tu

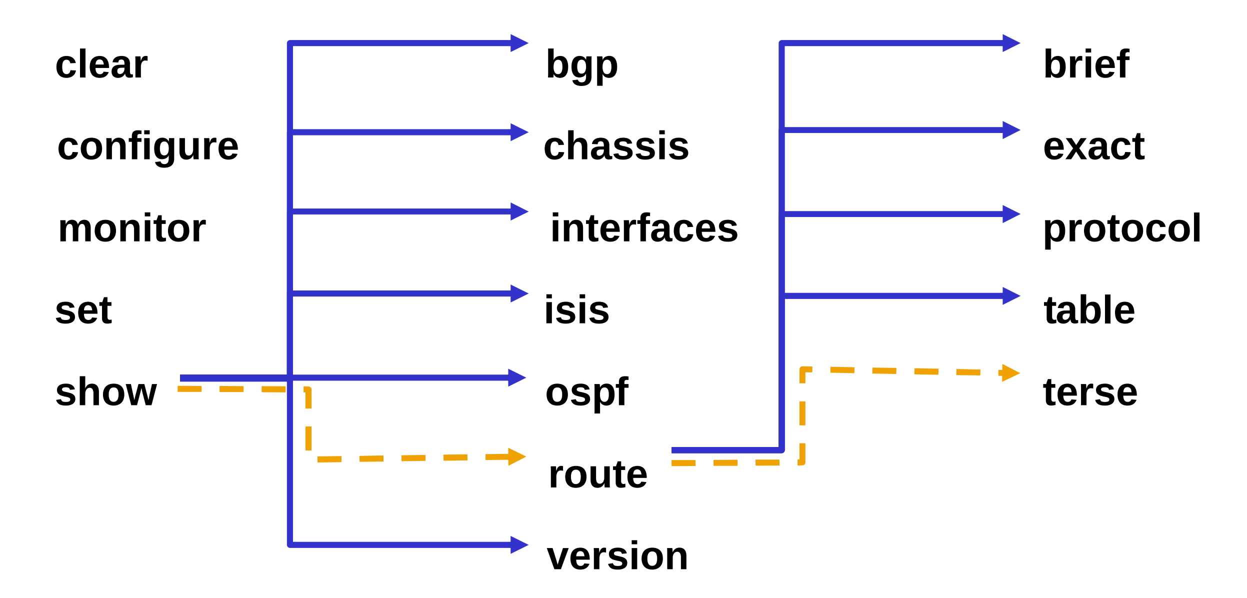


图 2-2

（2）主要命令

admin@RSR> ?

Possible completions:

clear	**Clear information in the system**
configure	**Manipulate software configuration information**
file	**Perform file operations**
help	**Provide help information**
monitor	**Show real-time debugging information**
mtrace	**Trace multicast path from source to receiver**
ping	**Ping remote target**
quit	**Exit the management session**
request	**Make system-level requests**
restart	**Restart software process**
set	**Set CLI properties, date/time, craft interface message**
show	**Show system information**
ssh	**Start secure shell on another host**
start	**Start shell**

telnet **Telnet to another host**
test **Perform diagnostic debugging**
traceroute **Trace route to remote host**

（3）可以配合使用下列管道符号 |

|compare、| count、| display、| except、| find、| hold、| match、| no-more、| resolve、| save、| trim

（4）使用 set cli 命令设置 cli 环境

admin@HBLF-PA-RT01> **set cli ?**
Possible completions:
  complete-on-space   Toggle word completion on space
  directory   Set the CLI's working directory
  idle-timeout   Set the CLI maximum idle time
  prompt   Set the CLI command prompt string
  restart-on-upgrade   Set CLI to prompt for restart after a software upgrade
  screen-length   Set number of lines on screen
  screen-width   Set number of characters on a line
  terminal   Set terminal type

（5）编辑环境在 VT-100

Ctrl+P	Displays the previous line in the CLI history buffer and is equivalent to the Up arrow key.
Ctrl+N	Displays the next line in the CLI history buffer and is equivalent to the Down arrow key.
Ctrl+B	Moves the cursor back one character and is equivalent to the Left arrow key.
Ctrl+F	Moves the cursor forward one character and is equivalent to the Right arrow key.
Esc+B	Moves the cursor back one word at a time. The Esc key must be released and re-pressed for each keystroke.
Esc+F	Moves the cursor forward one word at a time. The Esc key must be released and re-pressed for each keystroke.
Ctrl+A	Moves the cursor to the beginning of the current command line.
Crtl+E	Moves the cursor to the end of the current command line.
Ctrl+W	Deletes the word to the left of the cursor.
Ctrl+X	Deletes the entire current command line.
Ctrl+L	Redraws the current command line

（6）空格键功能

root@lab2> **sh<space>ow i<space>**
'i' is ambiguous.

Possible completions:
igmp Show information about IGMP
interfaces Show interface information
isis Show information about IS-IS

root@lab2> **show i**

（7）？ 号命令提示

lab@omaha> **?**
Possible completions:
clear Clear information in the system
configure Manipulate software configuration information
file Perform file operations
help Provide help information
…

2.1.2 配置模式

（1） 进入配置模式

root@lab2> **configure**
Entering configuration mode
[edit]
root@lab2#

example:
root@lab2#**set system services ftp**

```
system {
    services {
        ftp;
    }
}
```

（2）配置模式层次

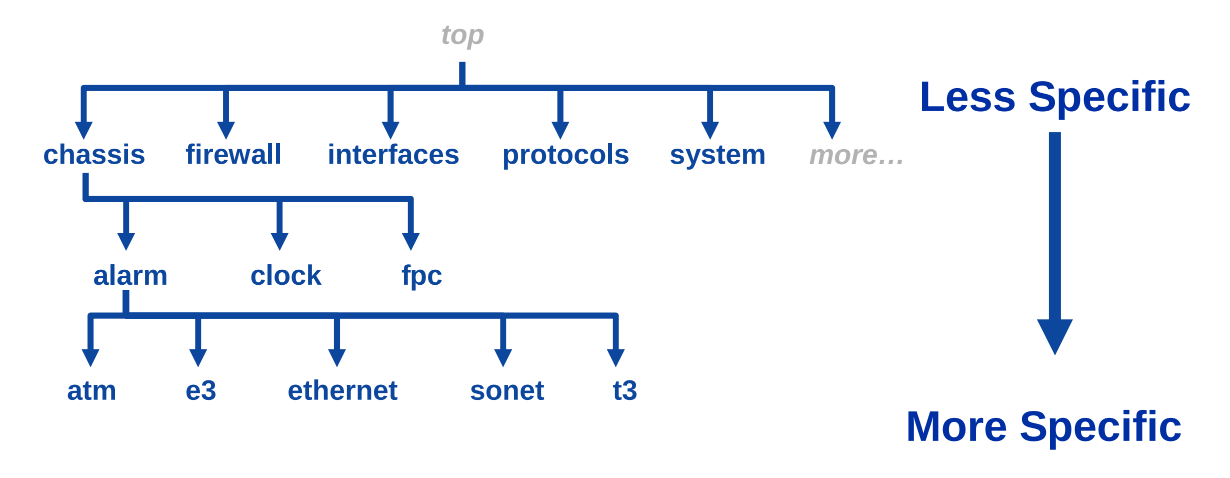


图 2 -3

（3）使用 edit 命令类似于 cd 命令

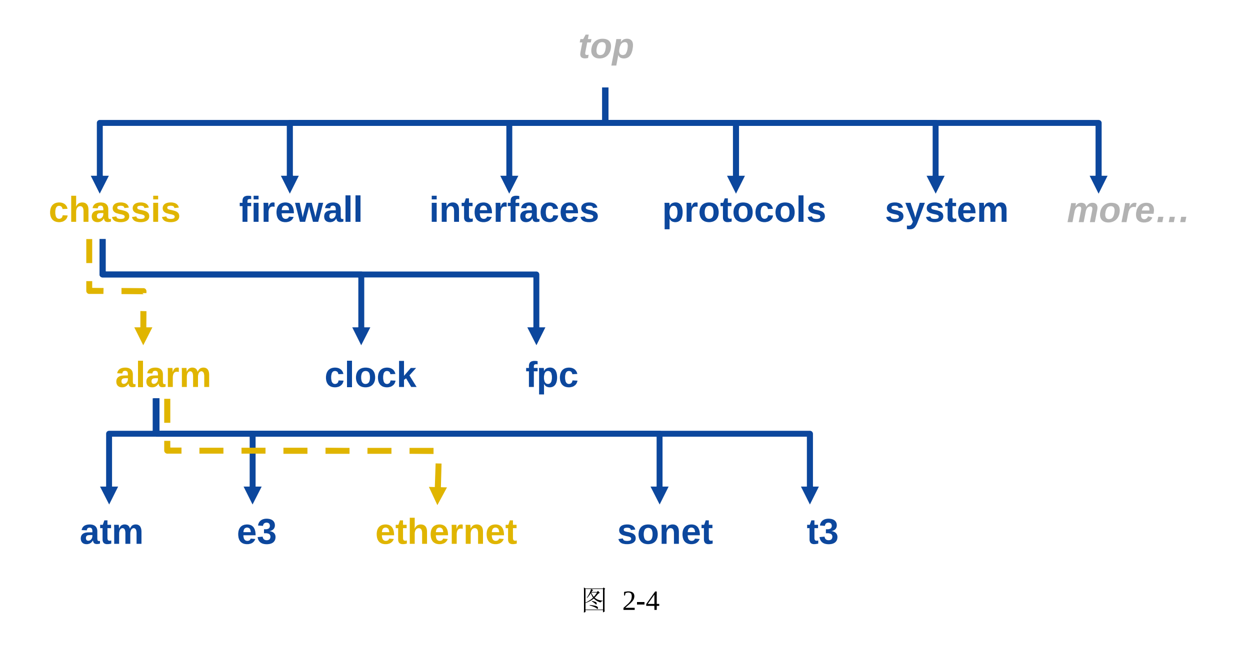


图 2-4

使用 edit chassis alarm ethernet 可以直接在 ethernet 这一级操作而不需要 set chassis alarm ethernet 操作。

（4） 使用 up 和 top 命令

user@host# **up**
[edit chassis alarm]
user@host# **top**
[edit]

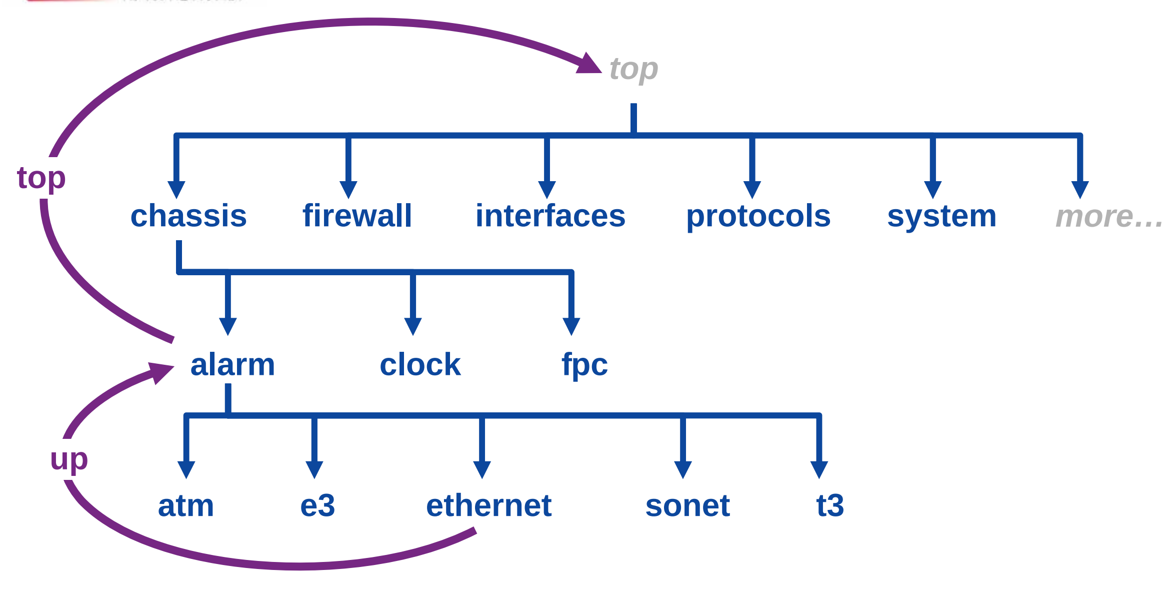


图 2－5

（5）显示 Candidate 配置

```
[edit]
user@host# show chassis alarm
sonet {
          lol red;
          pll yellow;
     }
[edit]
user@host# edit chassis alarm
[edit chassis alarm]
user@host# show
sonet {
          lol red;
          pll yellow;
     }
[edit chassis alarm]
```

（6）区分配置文件

- 改变现有配置

user@host# **set alarm sonet lol red**

user@host# **delete alarm sonet pll yellow**

- **比较 Candidate 与 active 配置的区别**

[edit chassis]

user@host# **show | compare**

alarm {

```
        sonet {
+           lol red
            los red;
-           pll yellow;
        }
    }
```

- 其它选项

user@host# **show | compare *filename***

user@host# **show | compare rollback *number***

（7）配置的差异更新

root@router# **show | compare**

```
[edit interfaces]
+   so-1/1/1 {
+       description "My new interface";
+       unit 0 {
+           family inet {
+               address 10.0.0.1/8;
+           }
+       }
+   }
[edit]
```

root@router# **show | compare | save /var/tmp/patch.cfg**

Wrote 9 lines of output to '/var/tmp/patch.cfg'

[edit]

root@router# **load patch /var/tmp/patch.cfg**

load complete

（8）删除配置

[edit]

user@host# **edit chassis alarm sonet**

[edit chassis alarm sonet]

user@host# **delete lol**

[edit chassis alarm sonet]

user@host# **delete los**

[edit chassis alarm sonet]

user@host#

（9）配置生效

- 使用 **commit** 是配置生效。**rollback** 恢复配置。

ser@host#**commit**

and-quit 如果配置成功退出配置模式

at 在定义的时间后 commit 配置

check 检测配置，不改变配置

confirmed 知道回滚等到下一个 commit 模式是 10 分钟

- **rollback** 恢复配置。**系统默认 9 个配置在 /config/ 3 个。file show /config/6** 个 **rollback** 或 **rollback 0** 就恢复到刚刚改变的配置

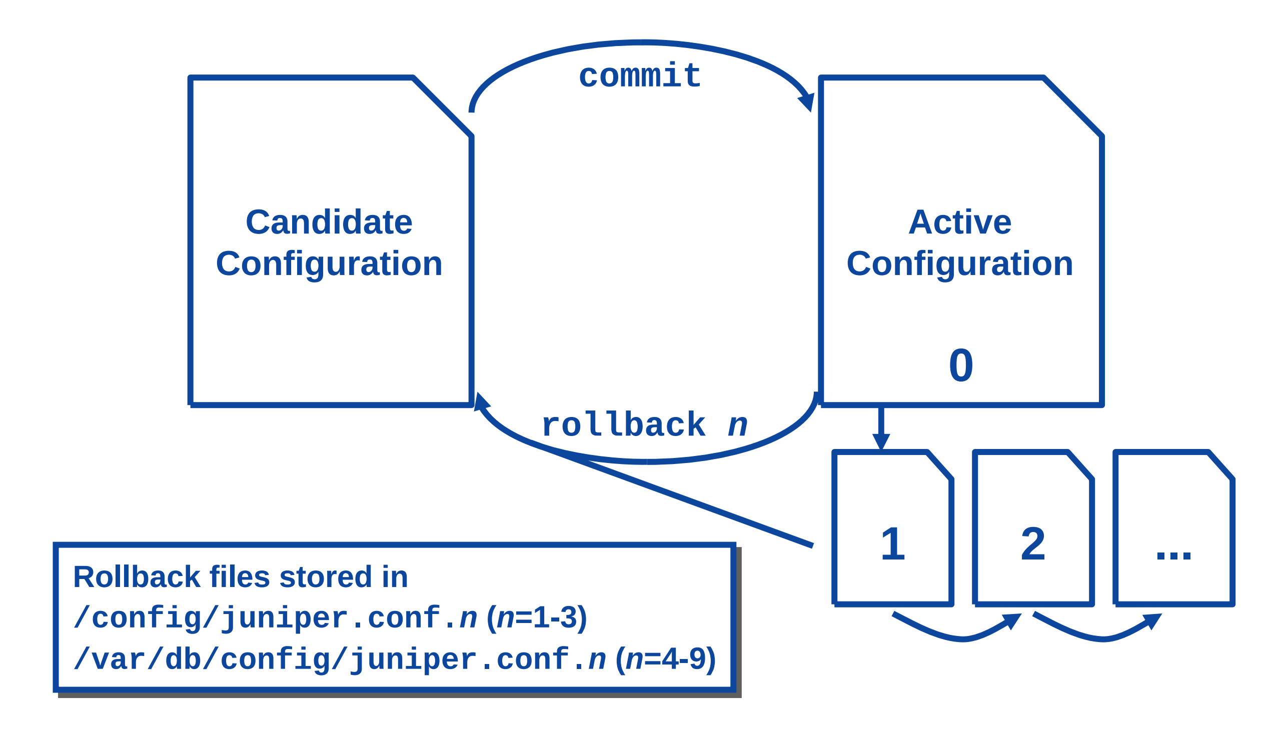


图 2－6

（10）配置模式层次改变。

- 使用 **exit** 退回到上一层配置模式
- 使用 **exit configuration-mode** 从任意层次退去配置模式

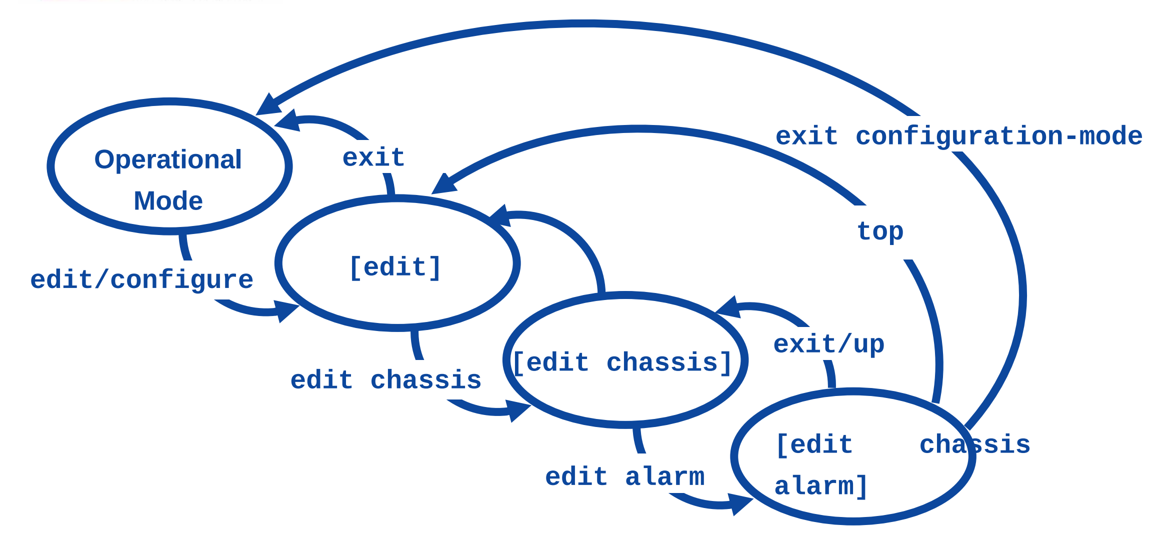


图 2-7

（11）保存加载配置文件

- 可以使用 **save** *filename* 在任何配置模式保持这个层次的配置内容到一个指定存放位置便于下一次 load。

[edit]
cli# **save *filename***
[edit]
cli#

- **load (replace | merge | override)**
  replace 取代现有的配置并且配置正确
  merge 合并配置
  override 取代现有的配置

user@host# **load merge** /var/db/config/juniper.conf.4

- 可以通过 **show system storage** 查看文件系统，或用 **file show** <*pathname*>来查看。

3. 系统管理和服务

3.1 Root 帐户

3.1.1 登陆

第一次登陆 router 用 root 用户，密码为空
login: **root**
Password:

--- JUNOS 7.3B1.1 built 2005-04-22 22:04:30 UTC
Terminal type? [vt100]

root% **cli**
root>

root> **configure**
Entering configuration mode

[edit]
Root#

3.1.2 配置 root 密码

(1)
root# **set root-authentication plain-text-password**
root# new password : star
root# retype new password: star
(2)密码以密文方式显示
root# show
    root-authentication {
        encrypted-password "1xavDeUe6$fNM6olGU.8.M7B62u05D6."; # SECRET-DATA
    }
（3）使用其它加密选项来加密 root 和 user 访问。
root# set **root-authentication encrypted-password** character-string
注意这种加密方式，手工输入时密码验证时通不过的。

3.2 管理 user 账号和访问

Junos 能支持配置大量的 user 访问，除了系统定义的默认访问权限外。还可以问用户定义权限。

3.2.1 user 帐户

root# **set login user** *robert*
root# **set login user robert authentication plain-text-password**

3.2.2 login classes

提供用户访问路由器的权限。可以使用系统提供的几个默认 classes 有四个。

Login class	**Permission Bits Set**
operator	clear、network、reset、trace、view
read-only	view
superuser	all
unauthorized	none

用户可以自己创建一个 class，系统提供的丰富的权限，也可以定义到具体用户使用的命令。
1， 用户自己定义一个组提供丰富的权限 **permissions 控制**
[edit system]
root@RSR04E-1# **set login class test permissions** *?*
Possible completions:

```
[                    Open a set of values
access               Can view network access configuration
access-control       Can modify network access configuration
admin                Can view user accounts
admin-coounts
all                  All permission bits turned on
clear                Can clear learned network information
configure            Can enter configuration mode
control              Can modify any configuration values
field    (debug) support
firewall             Can view firewall configuration
firewall-control     Can modify firewall configuration
floppy               Can read and write the floppy drive
interface            Can view interface configuratiodify interface configuration
maintenance          Can perform system maintenance (as wheel)
network              Can access the network
reset                Can reset and restart interfaces and processes
rollback             Can roll
```

routing Can view routing configuration
routing-control Can modify routing configuration
secret Can view secret configuration
secret-control Can modify secret configuration
security Can view security configuration
security-control Can modify security configuration
shell Can start a local shell
snmp figuration
snmp-control Can modify SNMP configuration
system Can view system configuration
system-control Can modify system configuration
trace Can view trace file settings
trace-control view Can view current values and statistics
view-configuration Can view all configuration (not including secrets)

[edit system]

Example : root@RSR04E-1# **set login class test permissions** *[clear network reset trace view configure interface-control]*

2， 用户自己定义一个允许特定命令的 class

root@RSR04E-1#**set login class test permissions** *[clear network reset trace view configure interface-control]* **allow-commands** *"system request reboot"* **deny-commands** *"(show system | show chassis | show version)"*

3, session timeout

root@RSR04E-1# s**et login class** class-name **idle-timeout** minutes;

4, 你也可以用 RADIUS 和 TACACS+ 来验证用户身份。

3.3 services and processes

3.3.1 设置 host name 、dns、domain-name

- 设置 Hostname

root@RSR04E# set system host-name RSR04E-1

[edit]
root@RSR04E# commit
commit complete

[edit]
root@RSR04E-1#

- 设置 DNS 、domain-name

[edit system]
root@RSR04E-1#**set name-server** 202.102.0.20

[edit system]
root@RSR04E-1#**set domain-name** RSR.star-net.cn

3.3.2 设置 Telnet、SSH

- 启动 telnet 服务

Syntax:
[edit system services]

telnet {
<connection-limit *limit*>;
<rate-limit *limit*>;
}
Example :
root@RSR04E-1# **set services telnet**

- 启动 ssh 服务

Syntax：
[edit system services]

ssh {
<connection-limit *limit*>;
protocol-version [*versions*]; /*** 版本参数 v1 、v2 ****/
<rate-limit *limit*>;
root-login (allow | deny | deny-password); /*** 是否允许root 用户ssh 方式登录 ****/
}

Example :
1,新建一个用户 admin
root@RSR04E-1# **edit system login**

[edit system login]
root@RSR04E-1# **set user admin class super-user authentication plain-text-password**
New password:*star123*
Retype new password:*star123*

[edit system login]
root@RSR04E-1# **top**

[edit]

2，启用 ssh 服务，协议指定版本 2，注意 ssh 版本 1 和版本 2 不兼容。

root@RSR04E-1# **edit system services**

[edit system services]

root@RSR04E-1# **set ssh protocol-version v2**

root@RSR04E-1# **commit**

commit complete

3， 用 secure CRT ssh 登陆

（1） name 任意取名，protocol：ssh2；port：ssh ；默认端口 22；username：admin；authentication（验证方式） ：password 方式；

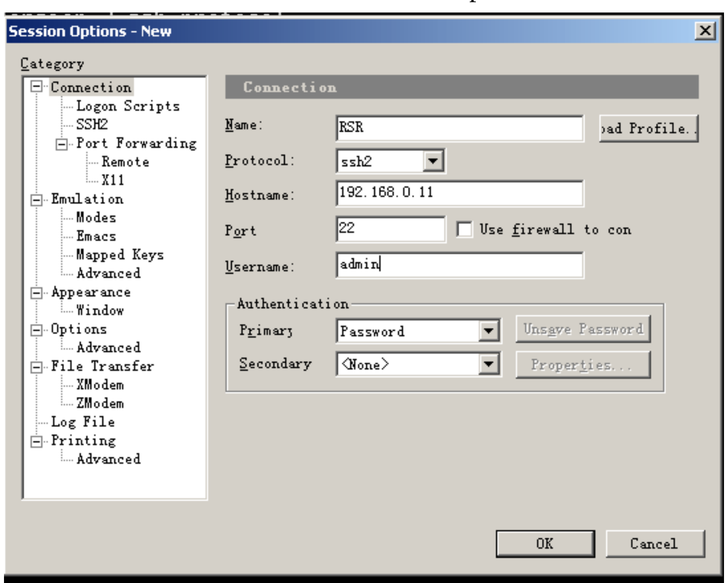


图 3－1

(2)选择 一或二都可以。

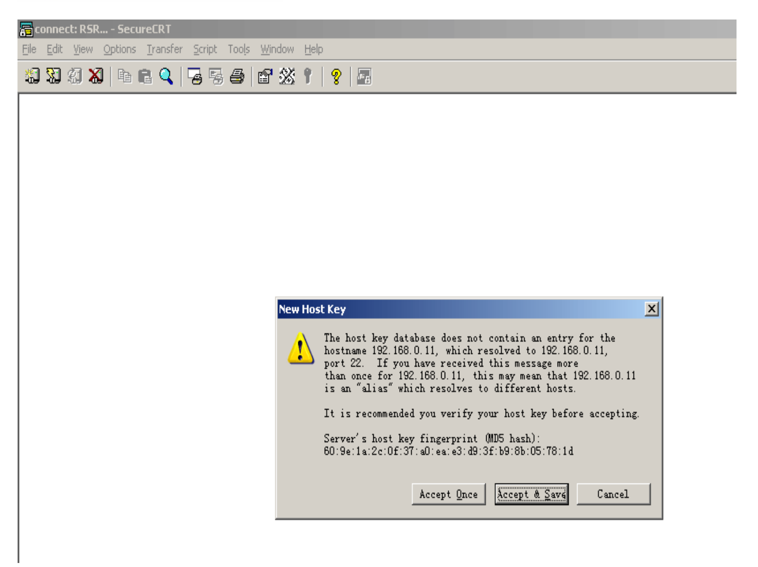


图 3-2

(3)输入 login password 就是 star123

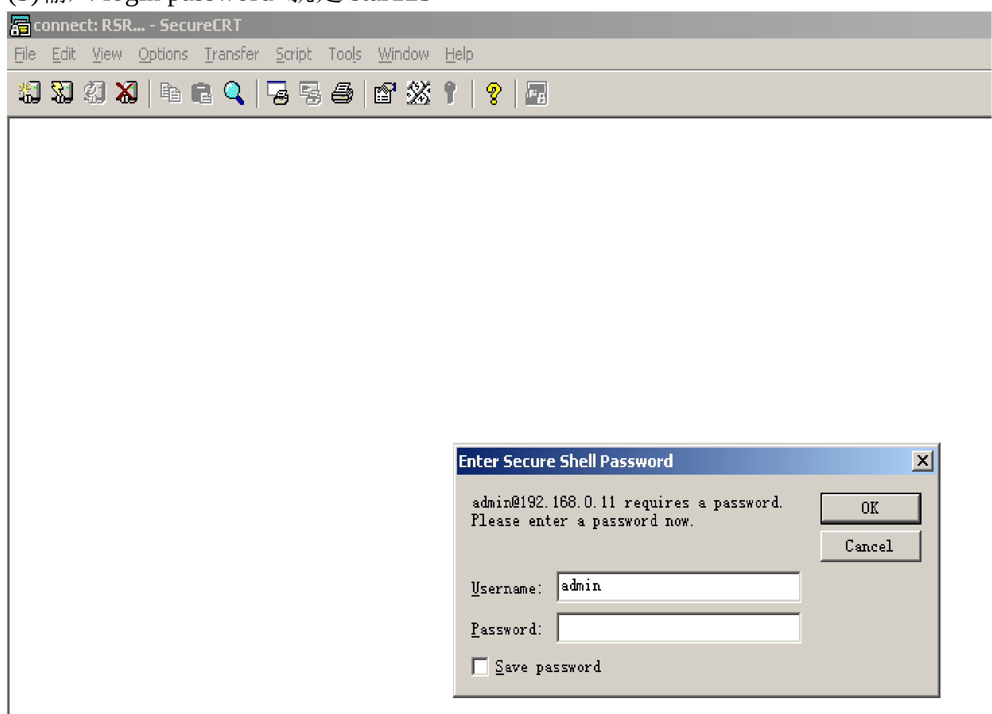


图 3 - 3

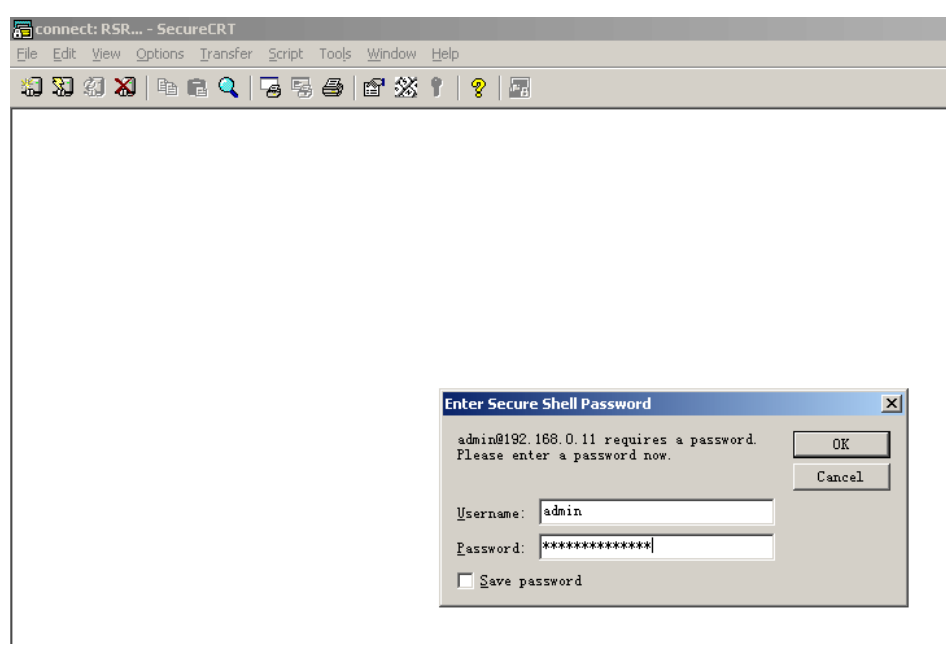


图 3-4

(4)登陆成功。

--- JUNOS 7.3B1.1 built 2005-04-22 22:04:30 UTC
admin@RSR04E-1>

（5）在 RSR 上使用 ssh 客户端方式登录
root@RSR04E-1> **ssh v2** admin@192.168.0.12 /*** admin 是用户名 ****/
admin@192.168.0.12's password: ********
--- JUNOS 7.2R1.7 built 2005-04-22 02:03:37 UTC
admin@RSR04E-2>

3.3.2 设置 FTP

（1）RSR 提供 ftp 服务用来用户 upload、 download JUNOS ，升级使用。
Syntax：

```
[edit system services]
    ftp {
        <connection-limit limit>;
        <rate-limit limit>;
    }
```

Example root@RSR04E-1# set ftp

[edit system services]
root@RSR04E-1#
（2） 如果需要使用 RSR 来做 ftp 客户端只有进入 unix shell 模式。不建议这样使用。
root@RSR04E-1>**start shell csh**

root@RSR04E-1% ftp 192.168.0.12
Connected to 192.168.0.12.
220 RSR04E-2 FTP server (Version 6.00LS) ready.
Name (192.168.0.12 : root): admin
331 Password required for admin.
Password:
230 User admin logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>

3.3.3 设置时间和时区、NTP （network time protocol）

- 设置 system 时间

operation mode：

set date *YYYYMMDDHHMM.ss*

通过 show system uptime 查看

root@RSR04E-1> **show system uptime**
Current time: 2005-08-15 08:05:52 UTC
System booted: 2005-08-15 05:34:58 UTC (02:30:54 ago)
Protocols started: 2005-08-15 05:35:40 UTC (02:30:12 ago)
Last configured: 2005-08-15 06:52:54 UTC (01:12:58 ago) by root
 8:05AM up 2:31, 1 user, load averages: 0.00, 0.00, 0.00

- 设置 Date and Time from NTP Servers

user@host> set date ntp "200.49.40.1 129.127.28.4"

- 设置时区

[edit system]
set time-zone time-zone

Example:
[edit system]
set time-zone America/New_York

- NTP的工作原理

N T P所针对的基本问题如图所示。两台路由器A和B通过串口相连，它们都有自己独立的系统时钟，问题是怎么样来实现各自系统时钟的自动同步？

我们假设：

• 在路由器A和B的系统时钟同步之前，路由器A的时钟设定为10:00 a.m.，路由器B的时钟设定为11:00 a.m.。

• 以路由器B为时间服务器(time server)，即路由器A将使自己的时钟与路由器B同步。

• 数据包在路由器A和B之间单向传输所需的时间为1秒

• 数据包在路由器B内部进行处理的时延为1秒。

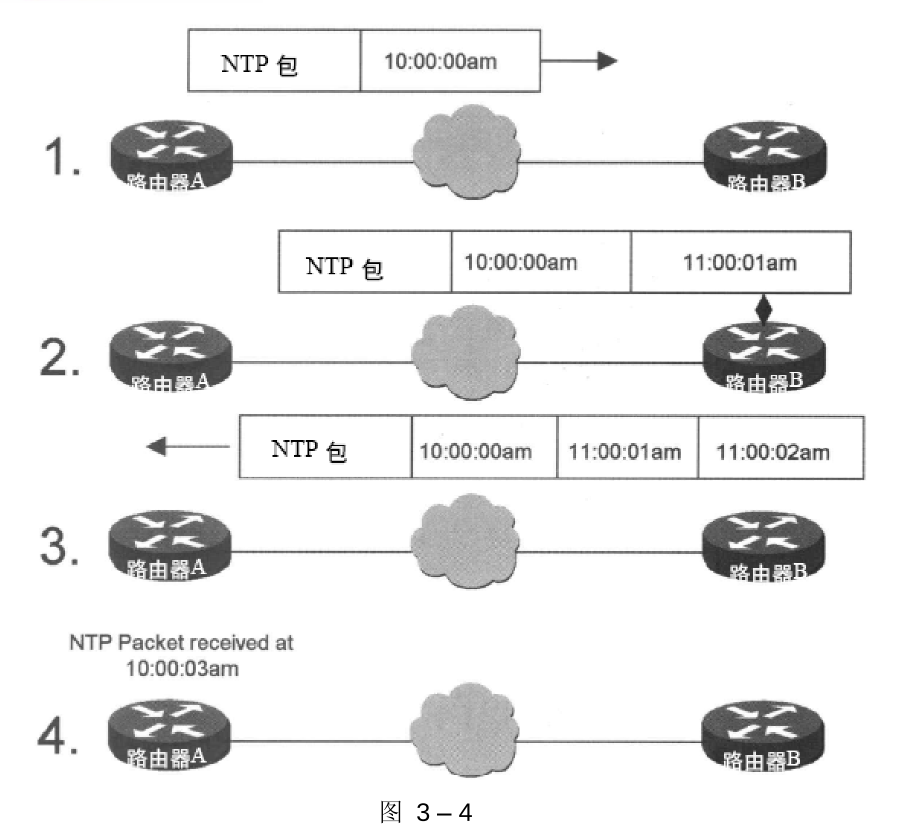


图 3 – 4

系统时钟同步的工作过程如下：

1) 路由器A发送一个N T P包给路由器B，该包带有它离开路由器A的时间戳，该时间戳为10:00:00 a.m.。

2) 当此N T P包到达路由器B时，路由器B加上自己的时间戳，该时间戳为11 : 0 0 : 0 1 a . m .。

3) 当此N T P包离开路由器B时，路由器B再次加上自己的时间戳，该时间戳为11:00:02 a.m.。

4) 当路由器A接收到该响应包时，加上一个新的时间戳，该时间戳为10:00:03 a.m.。

至此，路由器A已拥有足够的信息以计算两个重要的参数：

• NTP数据包来回一个周期的时延。

• 路由器A和路由器B的时钟差。

于是路由器A能够设定自己的时钟与路由器B同步。

应当注意这只是N T P工作原理的一个粗略描述，在R F C 1 3 0 5规范中，N T P使用复杂的算法来确保时钟同步的精确性。

● 设置NTP （JUNOS RFC 1305[8] version3）

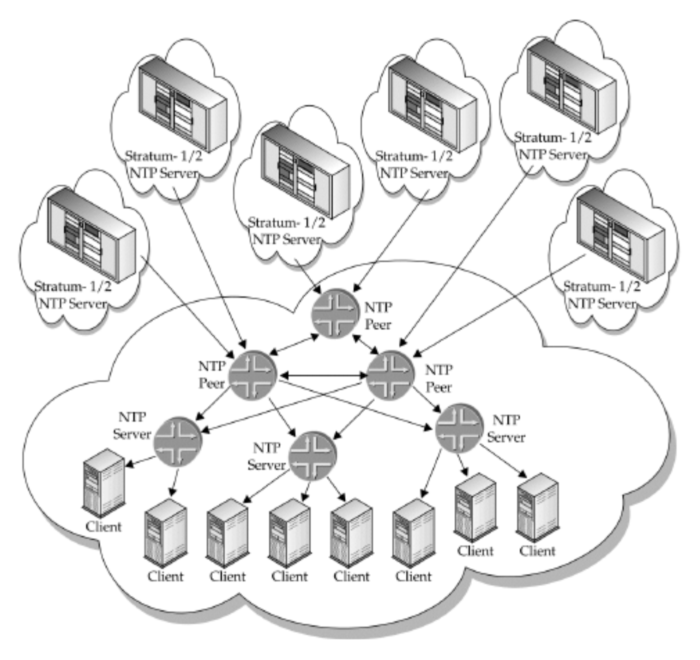


图 3 – 4

(1) JUNOS 支持实现 NTP 功能

Configuring the NTP Boot Server

Specifying a Source Address for an NTP Server

Configuring the NTP Time Server and Time Services

Configuring NTP Authentication Keys

Configuring the Router to Listen for Broadcast Messages

Configuring the Router to Listen for Multicast Messages

(2) Syntax：

```
[edit system]
Syntax
    ntp {
        authentication-key number type type value password;
        boot-server address;
        broadcast <address> <key key-number> <version value> <ttl value>;
        broadcast-client;
        multicast-client <ady key-number> <version value> <prefer>;
        server address <key key-number> <version value> <prefer>;
        source-address source-address;
        trusted-key [ key-numbers ];
    }
```

（3） 配置 ntp 客户模式

在客户端模式本地 router 能同步时间到远程 system。而远程 system 不能同步本地 router

[edit system ntp]

Syntax：

server *address* <key *key-number*> <version *value*> <prefer>;

authentication-key *key–number* type *type* value *password*;

boot-server *address*;

trusted-key [*key-number*s];

example :

[edit system ntp]

authentication-key 1 type md5 value *"9EgfcrvX7VY4ZEcwgoHjkP5Q3CuREyv87";*

boot-server *10.1.1.1*;

server *10.1.1.1* **key** 1 **prefer**;

trusted-key 1;

（4） 配置 Symmetric Active 模式

在 symmetric active 模式本地 router 和远程 system 都能同步其它时间源。

[edit system ntp]

peer *address* <key *key-number*> <version *value*> <prefer>

（5） 配置 NTP server 模式

本地router 运行一个ntp server

Syntax：

[edit system ntp]

authentication-key *key–number* type *type* value *password*;

server *address* <key *key-number*> <version *value*> <prefer>;

trusted-key [*key-numbers*];

example

[edit system ntp]

authentication-key *1* **type** *md5* **value** "9txERuBEreWx-wtuLNdboaUjH.T3AtOESe";

server *172.17.17.27.46* **prefer**;

trusted-key *1*;

（6） 验证NTP

Lab@RSR> show ntp status

Lab@RSR>show ntp associations

通过以上命令来查看ntp services 同步状态。

3.3.4 配置 SNMP

JUNOS 支持 SNMP version 3

(1) 配置 SNMP Agent 代理
Syntax

```
    community community-name {
        authorization authorization;
        clients {
            address restrict;
        }
        view view-name;
    }
```

Example:
admin@RSR04E-1# **set snmp community** *tester* **authorization read-only clients** *192.168.0.1/24*
admin@RSR04E-1# **set snmp community** *tester* **authorization read-only clients** 10.10.10.0/24 restrict

```
snmp {
     community tester {
         authorization read-only;
         clients {
             192.168.0.1/24;
             10.10.10.0/24 restrict；/****不允许 10.10.10.0/24 网段的主机访问 router snmp ****/
         }
     }
```

(2) 配置 SNMP trap-group

Syntax

```
    trap-group group-name {
        categories [ categories ];      /****  配置 trap 类型   ****/
        destination-port <port-number>;
        targets {
            address;                    /***  地址   ****/
        }
        version (all | v1 | v2);        /****   指定版本  *******/
    }
```

Example:

```
Snmp{
    trap-group tester {
            categories {
```

```
            link;      /***  路由器主动 trap  到指定主机    ***/
        }
         targets {
         192.168.0.1;
        }
    }
```

（2） 配置查看 SNMP 状态

admin@RSR04E-1# show snmp statistics

3.3.5 诊断工具 ping、traceroute、syslog、

1，使用 ping 测试网络连通性

- root@RSR04E-1>ping <address>

通过 ctrl+d 结束 ping 。

Example:

```
admin@RSR04E-1> ping 192.168.0.1
PING 192.168.0.1 (192.168.0.1): 56 data bytes
64 bytes from 192.168.0.1: icmp_seq=0 ttl=128 time=0.556 ms
64 bytes from 192.168.0.1: icmp_seq=1 ttl=128 time=0.324 ms
64 bytes from 192.168.0.1: icmp_seq=2 ttl=128 time=0.523 ms
^C
--- 192.168.0.1 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.324/0.468/0.556/0.102 ms
```

- ping options: Do-Not-Fragment and Size

admin@RSR04E-1> ping 192.168.0.1 do-not-fragment size 1400

- ping options: Record route

admin@RSR04E-1> ping 192.168.0.1 record-route

- ping options: detail

2, 使用 traceroute 命令

admin@RSR04E-1>traceroute   www.yahoo.com.cn

选项：noresolve、source、TTL、

3, Syslog

Junos 可以把 syslog 发送到 console（控制台端口），日志-服务器，本地硬盘。

- **facilities and Severity levels**

JUNOS system logging Facilities (logging 产生来源)

Facility	Type of Event or Error
any	All (messages from all facilities)
authorization	Authentication and authorization attempts
change-log	Changes to the JUNOS configuration
conflict-log	Configuration that is inconsistent with routing platform hardware
daemon	Actions performed or errors encountered by various system processes
firewall	Packet filtering actions performed by a firewall filter
ftp	Actions performed or errors encountered by the FTP process
interactive-commands	Commands issued at the JUNOS command-line interface (CLI) prompt or by a JUNOScript client application
kernel	Actions performed or errors encountered by the JUNOS kernel
pfe	Actions performed or errors encountered by the Packet Forwarding Engine
user	Actions performed or errors encountered by various user-space processes

System Log Message Severity Levels（logging 产生级别）

Severity Level	Description
any	Includes all severity levels
none	Disables logging of the associated facility to a destination
emergency	System panic or other condition that causes the routing platform to stop functioning
alert	Conditions that require immediate correction, such as a corrupted system database
critical	Critical conditions, such as hard drive errors
error	Error conditions that generally have less serious consequences than errors in the emergency, alert, and critical levels
warning	Conditions that warrant monitoring
notice	Conditions that are not errors but might warrant special handling
info	Events or nonerror conditions of interest

Syntax :

```
syslog {
    archive {
        files number;
        size size;
        (world-readable | no-world-readable);
    }
    console {
        facility severity;
    }
    file filerity;
        explicit-priority;
        match "regular-expression";
        archive {
            files number;
            size size;
            (world-readable | no-world-readable);
        }
    }
    host (hostname | other-routing-engine | scc-master) {
        facility severity;
        explicit-priority;
        facility-overrde facility;
        log-prefix string;
        match "regular-expression";
  ce-address;
    time-format (year | millisecond | year millisecond);
    user (username | *) {
        facility severity;
        match "regular-expression";
    }
}
```

Syslog Files

- 指定 log 类型，存放文件名

admin@RSR04E-1# edit system

[edit system]

admin@RSR04E-1# **set syslog file** messages *any notice*

- 指定 log 日志文件尺寸，文件个数

默认 log 日志文件 128k，一共存放 10 个

admin@RSR04E-1# edit system

[edit system]

admin@RSR04E-1# s**et syslog archive** *size* 1m *files* 20

文件大小 1M，文件存放 20 个。

- 指定 syslog server 标准使用 udp 514 端口。

[edit system]

admin@RSR04E-1# **set syslog host** 192.168.0.1 *any alert*

使用标准的local*X* facility name

admin@RSR04E-1# **set syslog host 192.168.0.1 facility-override** *local0*

4， Traceoptions

在每个配置模式下定义 log

admin@RSR04E-1# **set traceoptions file** *interface.log* **size** *1m* **files** *10*

admin@RSR04E-1# **set traceoptions flag** *change-events*

example

```
interfaces {
    traceoptions {
        file interface.log size 1m files 10;
        flag change-events;
    }
)
```

查看

admin@RSR04E-1>Show log

Log 文件路径在/var/log/

4，interface 配置和控制

- **RSR04E、RSR08E 接口卡，每 1 个 CFEB 可以插入 4 个 PIC。**

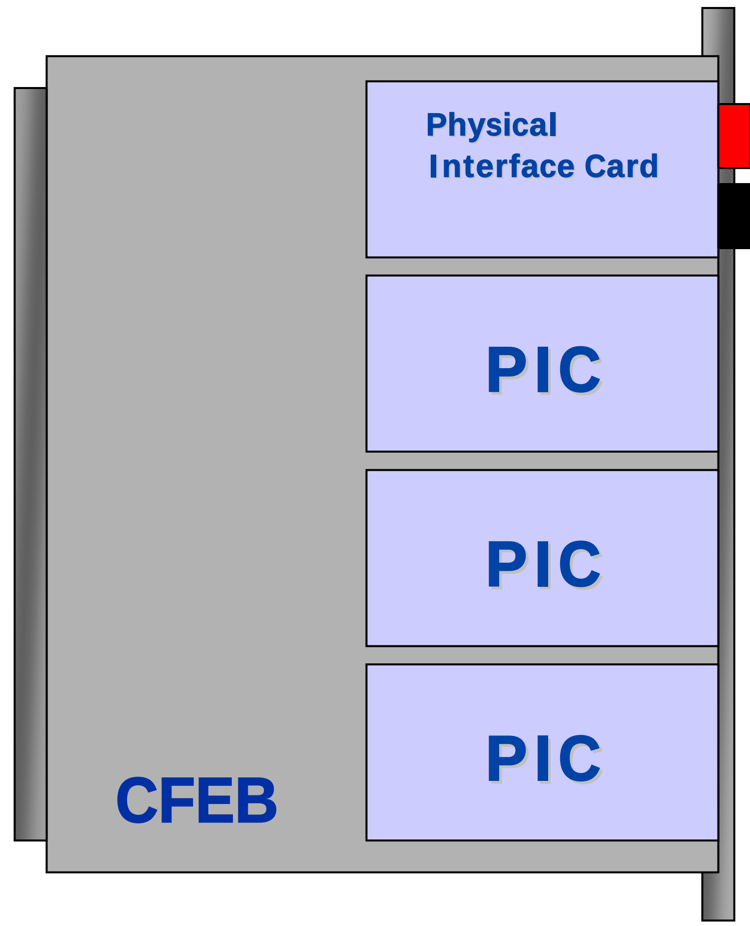


图 4-1

4.1 标准接口命名规则

设备启动请使用 **show interfaces terse** 来查看所以物理端口。

Interface type-CFEB # / PIC Slot＃ ／Port #

- **Interface Media Type**
  - **at—ATM over SONET/SDH ports**
  - **e1—E1 ports**
  - **e3—E3 ports**
  - **fe—Fast Ethernet ports**

- so—SONET/SDH ports
- t1—T1 ports
- t3—DS-3 ports
- ge—Gigabit Ethernet ports
- ae—Aggregated Ethernet ports

CFEB slot number

RSR 04E 只有一个 CFEB ，因此 CFEB number 0
RSR 08E 有两个 CFEB，上面是 0，下面为 1，如图：

	0
	1

图 4-2

PIC slot number

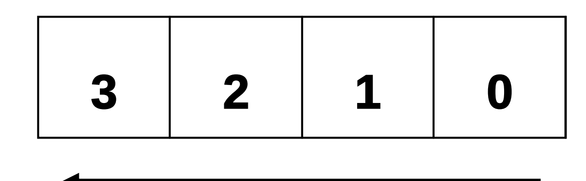


图 4 -3

Interface name

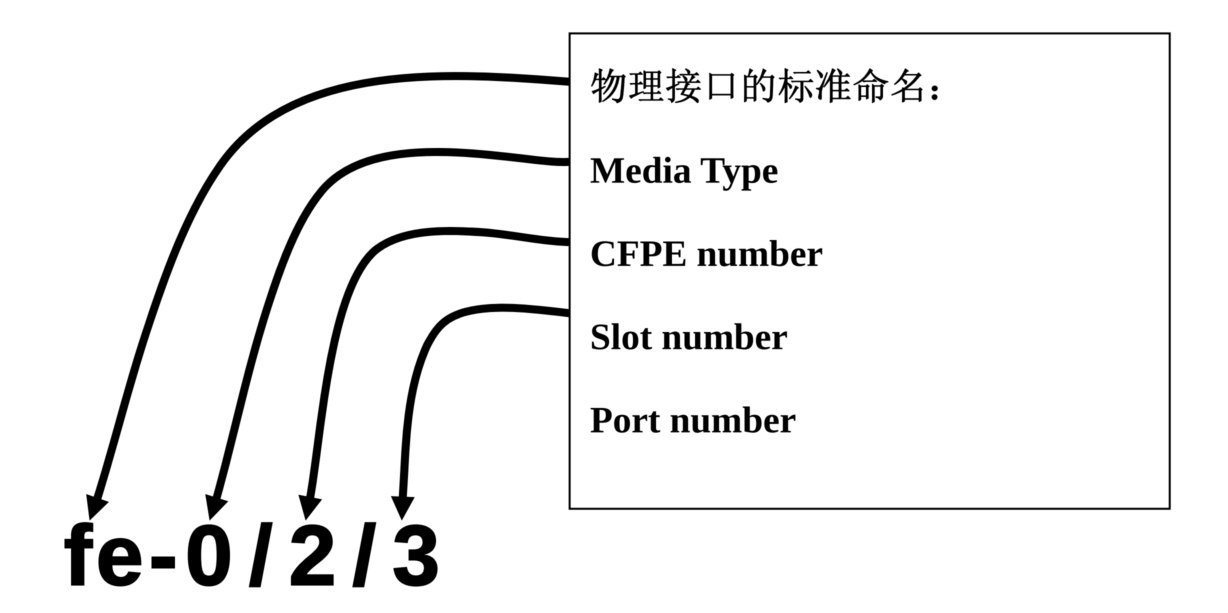


图 4 -4

- **Ethernet 逻辑端口**

  fe-0/2/3.0

- **永久接口**

fxp0 路由引擎上的 MGMT （Out-of-band management interface ）网管接口，没有路由功能。

fxp1 连接路由引擎和转发引擎的接口。请不要配置和操作这个接口。

- **Interface 参数**

物理接口参数

- **Clocking**
- **Scrambling**
- **Frame check sequence (FCS)**
- **Maximum transmission unit (MTU)**
- **Data Link Layer protocol, Keepalives**
- **Diagnostic characteristics**
  - **Local, remote, and facility loopback**
  - **BERT**

逻辑接口参数

- **Protocol family (Internet, ISO, MPLS)**
- **Addresses (IP address, ISO NET address)**
- **Virtual circuits (VCI/VPI, DLCI)**
- **Other characteristics**

标准配置层次

Syntax:

```
interfaces {
    interface-name {
        physical-properties;
        […]
        unit unit-number {
            logical-properties;
            […]
        }
    }
}
```

- **配置物理接口**

admin@RSR04E-1> configure
Entering configuration mode
admin@RSR04E-1# set interfaces fe-0/0/0 mtu 1514

[edit]
admin@RSR04E-1# commit
commit complete

[edit]

- **配置逻辑接口**

配置分装类型、子接口。
admin@RSR04E-1# set fe-0/0/0 unit 0 family inet address 192.168.11.1/24

- **关于 unit number**

1，每一个逻辑接口有一个 **unit** 号
　　这个号能代表相同的 ATM vc、frame- rely DLCI 号
2，有些物理接口有只有一个可能的逻辑接口和一个 **unit** 号，所以 **unit** 号配置为 **0**
3，多个协议地址是支持单个逻辑 **unit** 号

4.2 配置 Ethernet interface : GE and FE

RSR 支持 GE、FE、支持 VLAN tagging、配置 MTU 等

4.4.1 MAC 地址设置

每个路由器有 1024 个地址块，16 个保留给系统用。其中两个给 fxp1、fxp0 使用，其余 1008 可以分配给逻辑接口。可以通过手工设置 MAC 地址 set fe-0/0/0 mac。 通过 show chassis mac 来查看地址块

admin@RSR04E-1> **show chassis mac-addresses**
MAC address information:
　Public base address　　00:12:1e:01:30:00

Public count 1008
Private base address 00:12:1e:01:33:f0
Private count 16

4.4.2 VLAN Tagging

RSR 支持 802.1Q VLAN tagging. Vlan IDs 0 到 4095

Example:

1, RSR04E 与 RG—6806E trunk 802.1Q FE trunk

(1) RSR config

```
interfaces {
    fe-0/0/0 {
        description to-RG6808-4/5;
        vlan-tagging;
        unit 11 {
            vlan-id 11;
            family inet {
                address 10.10.11.2/24;
            }
        }
        unit 12 {
            vlan-id 12;
            family inet {
                address 10.10.12.2/24;
            }
        }
    }
```

(2) RG-6806E

```
interface GigabitEthernet 4/5
 speed   100
 duplex full
 description "to-RSR04E-fe-0/0/0"
switchport mode trunk
```

2, RSR04E 与 RG—6806E trunk 802.1Q GE trunk

（1）RG-6806E 配置

```
    interface GigabitEthernet 4/9
```

```
   medium-type fiber
   speed    1000
   description "to-RSR04E-ge-1/3/0"
  switchport mode trunk
```

(2) RSR04E 配置

```
interfaces {
    ge-1/3/0 {
        description to-RG-6808E;
        vlan-tagging;
        unit 13 {
            vlan-id 13;
            family inet {
                address 10.10.13.2/24;
            }
        }
        unit 14 {
            vlan-id 14;
            family inet {
                address 10.10.14.2/24;
            }
        }
    }
```

4.2.3 Source Filtering

过滤 mac 地址，

```
interfaces {
    ge-1/3/0 {
        description to-RG-6808E;
        vlan-tagging;
        gigether-options {
            source-filtering;        /****  设定源地址过滤  *****/
            source-address-filter {
                00:90:69:6e:b8:01;     /*****    过滤具体地址  *****/

            }
        }
    }
}
```

4.3 配置 VRRP （RFC 2338）

4.3.1 工作原理

一个 VRRP 路由器有唯一的标识：VRID，范围为 0—255。该路由器对外表现为唯一的虚拟 MAC 地址，地址的格式为 00-00-5E-00-01-[VRID]。主控路由器负责对 ARP 请求用该 MAC 地址做应答。这样，无论如何切换，保证给终端设备的是唯一一致的 IP 和 MAC 地址，减少了切换对终端设备的影响。

VRRP 控制报文只有一种：VRRP 通告(advertisement)。它使用 IP 多播数据包进行封装，组地址为 224.0.0.18，发布范围只限于同一局域网内。这保证了 VRID 在不同网络中可以重复使用。为了减少网络带宽消耗只有主控路由器才可以周期性的发送 VRRP 通告报文。备份路由器在连续三个通告间隔内收不到 VRRP 或收到优先级为 0 的通告后启动新的一轮 VRRP 选举
在 VRRP 路由器组中，按优先级选举主控路由器，VRRP 协议中优先级范围是 0—255。若 VRRP 路由器的 IP 地址和虚拟路由器的接口 IP 地址相同，则称该虚拟路由器作 VRRP 组中的 IP 地址所有者；IP 地址所有者自动具有最高优先级：255。优先级 0 一般用在 IP 地址所有者主动放弃主控者角色时使用。可配置的优先级范围为 1—254。优先级的配置原则可以依据链路的速度和成本、路由器性能和可靠性以及其它管理策略设定。主控路由器的选举中，高优先级的虚拟路由器获胜，因此，如果在 VRRP 组中有 IP 地址所有者，则它总是作为主控路由的角色出现。对于相同优先级的候选路由器，按照 IP 地址大小顺序选举。VRRP 还提供了优先级抢占策略，如果配置了该策略，高优先级的备份路由器便会剥夺当前低优先级的主控路由器而成为新的主控路由器。

为了保证 VRRP 协议的安全性，提供了两种安全认证措施：明文认证和 IP 头认证。明文认证方式要求：在加入一个 VRRP 路由器组时，必须同时提供相同的 VRID 和明文密码。适合于避免在局域网内的配置错误，但不能防止通过网络监听方式获得密码。IP 头认证的方式提供了更高的安全性，能够防止报文重放和修改等攻击。

4.3.2 RSR 实现

Syntax:

```
vrrp-group group-number {
    (accept-data | no-accept-data);
    advertise-interval seconds;
    authentication-type authentication;
    authentication-key key;
    fast-interval milliseconds;
    (preempt | no-preempt) {
        hold-time seconds;
    }
    priority number;
    track {
        interface interface-name priority-cost cost;
```

```
        }
        virtual-address [ addresses ];
    }
```

Exaple：

Router A

```
fe-0/0/1 {
    unit 0 {
        family inet {
            address 10.10.11.2/24 {
                vrrp-group 10 {
                    virtual-address 10.10.11.1;
                    priority 105;
                    accept-data;
                }
            }
        }
    }
}
```

Router B

```
fe-0/0/1 {
    unit 0 {
        family inet {
            address 10.10.11.3/24 {
                vrrp-group 10 {
                    virtual-address 10.10.11.1;
                    priority 150;
                    accept-data;
                }
            }
        }
    }
}
```

图 4 -6

admin@RouterA # **show vrrp detail**

Physical interface: fe-0/0/1, Unit: 0, Address: 10.10.11.2/24

Index: 65, SNMP ifIndex: 31, VRRP-Traps: disabled

Interface state: up, Group: 10, State: backup

Priority: 105, Advertisement interval: 1, Authentication type: none

Preempt: yes, Accept-data mode: yes, VIP count: 1, VIP: 10.10.11.1

Dead timer: 2.921s, Master priority: 150, Master router: 10.10.11.3

Virtual router uptime: 00:21:53

Tracking: disabled

admin@RouterB>**show vrrp detail**

Physical interface: fe-0/0/1, Unit: 0, Address: 10.10.11.3/24

Index: 65, SNMP ifIndex: 25, VRRP-Traps: disabled

Interface state: up, Group: 10, State: master

Priority: 150, Advertisement interval: 1, Authentication type: none

Preempt: yes, Accept-data mode: no, VIP count: 1, VIP: 10.10.11.1

Advertisement timer: 0.779s, Master router: 10.10.11.3

Virtual router uptime: 00:20:41, Master router uptime: 00:20:37

Virtual MAC: 00:00:5e:00:01:0a

Tracking: disabled

注意加入 **accept-data** 参数 允许相应对 **vip** 的 **ICMP** 响应。

如果你配置加密验证可能需要通过 **clear vrrp** 来初始化 **vrrp**。

4.4 Aggregated Interfaces

Junos 支持聚合 Ethernet 类型接口，必须是 full-duplex mode with VLAN tagging。Sonet 类型的接口。

4.4.1 实现步骤一：设置 chassis 支持具体类型的端口聚合。

set chassis aggregated-devices (Ethernet | sonnet) device-count [1 – 128]
example:
[edit]
admin@RSR04E-2# **set chassis aggregated-devices ethernet device-count** 1

4.4.2 步骤二：配置 ag0 和 参数

[edit interfaces ae0]
admin@RSR04E-2# **set vlan-tagging**

[edit interfaces ae0]
admin@RSR04E-2# **set unit 100 vlan-id 100**

[edit interfaces ae0]
admin@RSR04E-2# **set unit 100 family inet address 10.0.10.3/24**

[edit interfaces ae0]
admin@RSR04E-2# **set aggregated-ether-options minimum-links 2**   /***捆绑最少两条链路 ***/

4.4.3 步骤三：配置加入捆绑端口；

```
[edit interfaces]
admin@RSR04E-2# set fe-0/0/0 fastether-options 802.3ad ae0

[edit interfaces]
admin@RSR04E-2# set fe-0/0/1 fastether-options 802.3ad ae0
```

4.4.4 配置例子

（1）Example：

RSR04E-1　　ag0: 10.10.11/24　　RSR04E-2

.1　　.2

= 200M

Fe-0/0/(2-3)　　Fe-0/0/(2-3)

图 4 -6

RSR04E-1

```
[edit]
admin@RSR04E-1> show
chassis {
        aggregated-devices {
                ethernet {
                        device-count 1;
                }
        }
}
interfaces {
        fe-0/0/2 {
                fastether-options {
                        802.3ad ae0;
                }
        }
        fe-0/0/3 {
                fastether-options {
                        802.3ad ae0;
                }
        }
ae0 {
            vlan-tagging;
            aggregated-ether-options {
                    minimum-links 2;
            }
            unit 100 {
                    vlan-id 100;
                    family inet {
                            address 10.10.11.1/24;
                    }
            }
     }
}
```

RSR04E-2 基本相同；

admin@RSR04E-1> **show interfaces ae0**
Physical interface: ae0, Enabled, Physical link is Up
Interface index: 142, SNMP ifIndex: 46
Link-level type: Ethernet, MTU: 1518, **Speed: 200mbps**, Loopback: Disabled, Source filtering: Disabled, Flow control: Disabled,
Minimum links needed: 2
Device flags : Present Running
Interface flags: SNMP-Traps 16384
Current address: 00:12:1e:01:33:f0, Hardware address: 00:12:1e:01:33:f0
Last flapped : Never
Input rate : 0 bps (0 pps)
Output rate : 0 bps (0 pps)

Logical interface ae0.100 (Index 69) (SNMP ifIndex 64)
Flags: SNMP-Traps 16384 VLAN-Tag [0x8100.100] Encapsulation: ENET2

Statistics	Packets	pps	Bytes	bps
Bundle:				
Input :	10	0	816	0
Output:	10	0	1006	0

Protocol inet, MTU: 1500
Flags: None
Addresses, Flags: Is-Preferred Is-Primary
Destination: 10.10.11/24, Local: 10.10.11.1, Broadcast: 10.10.11.255

2 Example

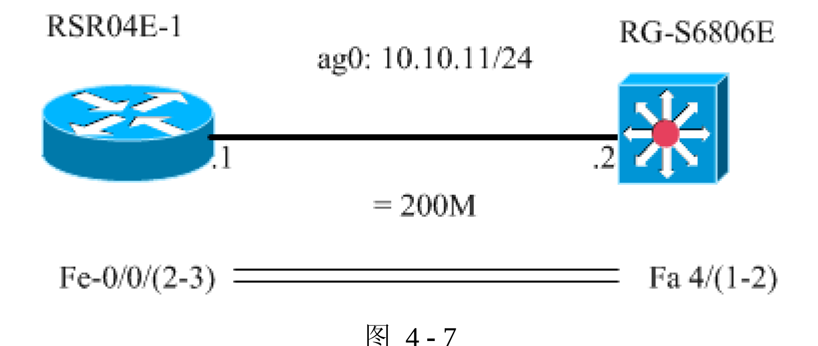


图 4 - 7

RG-6806E# show
interface AggregatePort 1
switchport mode trunk
!
interface GigabitEthernet 4/1

```
 port-group 1
 speed    100
 duplex full
!
interface GigabitEthernet 4/2
 port-group 1
 speed    100
 duplex full
!
interface Vlan 100
 ip address 10.10.11.2 255.255.255.0
!
```

4.5 Troubleshooting interface

4.5.1 monitor interface

可以通过 monitor interface <interface> 来动态查看一个接口
可以通过 monitor traffic interface 来监控

Example:

```
dmin@RSR04E-1> monitor traffic interface fe-0/0/1
verbose output suppressed, use <detail> or <extensive> for full protocol decode
Listening on fe-0/0/1, capture size 96 bytes

08:32:23.253227    In IP 10.10.11.3 > 224.0.0.18: VRRPv2-advertisement 20: vrid=10 prio=150 authtype=simple intvl=1
08:32:24.583251    In IP 10.10.11.3 > 224.0.0.18: VRRPv2-advertisement 20: vrid=10 prio=150 authtype=simple intvl=1
08:32:26.073227    In IP 10.10.11.3 > 224.0.0.18: VRRPv2-advertisement 20: vrid=10 prio=150 authtype=simple intvl=1
^C
3 packets received by filter
0 packets dropped by kernel
```

4.5.1 show interfaces fe-0/0/0 extensive

详细察看接口状态。

4.5.2 在 E-1 Loopback 测试物理链路使用

```
admin@RSR# set loopback ?
possible completions:
   local                    local loopback
   remote                   remote loopback
```

5. protocol-Independent Routing

5.1 静态路由协议

- 手工添加到路由表
- 下一跳必须有效
- 所以的配置在 routing-options 级别

Syntax:

```
[edit]
routing-options {
    static {
        defaults {
            static-options;
        }
        route destination-prefix {
            next-hop next-hop;
            qualified-next-hop address {
                metric metric;
                preference preference;
            }
            static-options;
        }
    }
}

Example:
[edit]
user@host# show
routing-options {
    static {
        route 0.0.0.0/0 next-hop 192.168.0.1;
    }
}
```

- Discard 、Reject 选项 ，

类似 cisco 把下一跳指到 null 0 接口，两者不同处 Reject 删除包并发出 ICMP "Destination Host Unreachable" 消息。Discard 删除包没有 ICMP 消息

```
routing-options {
  static {
    route 143.172.0.0/6 discard;
  }
```

```
}

routing-options {
        static {
            route 143.172.0.0/6 Reject;
        }
}
```

- qualified-next-hop 选项

qualified-next-hop 同一目的地，指定多个下一跳类似浮动路由。

5.2 Aggregated Routes 和 Generated Routes

用于把多条路由聚合成一个单一的路由加入到路由表。Aggregated、Generated 两者的区别下一跳属性不一样。默认 Aggregated 的下一跳是 reject，discard 是可选项；Generated 多用于路由最后匹配，或者浮动路由，下一条根据聚合的条件路由其中最匹配的来设置下一条，Discard 也是选项。

Syntax：

```
[edit]
routing-options {
    aggregate {
        defaults {
            aggregate-options;
        }
        route destination-prefix {
            policy policy-name;
            aggregate-options;
        }
    }
}
```

Example：

```
routing-options {
    aggregate {
        defaults {
            community 1:888;  /*** bgp 使用的属性   ***/
        }
        route 192.168.16.0/21;
        route 192.168.24.0/21 discard;
        }
    }
```

Syntax:

```
[edit]
routing-options {
   generate {
      defaults {
         generate-options;
      }
      route destination-prefix {
         policy policy-name;
         generate-options;
      }
```

Example:

```
routing-options {
    generate {
        defaults {
            metric 5;
        }
        route 172.16.64.0/20;
        route 172.16.80.0/20 discard;
        }
    }
```

5.3 Martian Routes

根据 Assigned Numbers Authority (IANA).定义

http://www.iana.org/assignments/ipv4-address-space.

The default list of martian routes in the JUNOS software is:

- Prefix bits of 0.0.0.0 /8 and more specific routes
- Prefix bits of 127.0.0.0 /8 and more specific routes
- Prefix bits of 128.0.0.0 /16 and more specific routes
- Prefix bits of 191.255.0.0 /16 and more specific routes
- Prefix bits of 192.0.0.0 /24 and more specific routes
- Prefix bits of 223.255.255.0 /24 and more specific routes
- Prefix bits of 240.0.0.0 /4 and more specific routes

Junos 默认是不可以使用这些转发这些地址的。但可以配置。

Syntax：

```
routing-options {
    martians {
        prefix/prefix-length match-type allow;
    }
}
```

Match-type exact、longer、orlonger、prefix-length-range、through、upto;

user@Riesling> **show route martians**

```
inet.0:
             0.0.0.0/0 exact -- allowed
             0.0.0.0/8 orlonger -- disallowed
             127.0.0.0/8 orlonger -- disallowed
             128.0.0.0/16 orlonger -- disallowed
             191.255.0.0/16 orlonger -- disallowed
             192.0.0.0/24 orlonger -- disallowed
             223.255.255.0/24 orlonger -- disallowed
             240.0.0.0/4 orlonger - disallowed
```

5.4 JUNOS 路由表

junos 提供多个路由表存放路由。用户可以自己地址路由表。系统默认路由表：

- inet.0 - store IPv4 unicast routes
- inet.1 - store IPv4 multicast routes
- inet.2 - store IPv4 multicast routes， Routes in the inet.2 table are used by multicast routing protocols to prevent routing loops
- inet.3 - The *inet.3* routing table contains the egress IP address of a MPLS label switched path (LSP).
- inet.4 - The *inet.4* routing table stores information learned using the Multicast Source Discovery Protocol (MSDP).
- inet6.0 - The *inet6.0* routing table contains IPv6 unicast routes
- mpls.0 - The *mpls.0* table is not actually a routing table but is instead a switching table. MPLS label values are stored in this table.
- bgp.l3vpn.0 - The *bgp.l3vpn.0* routing table stores routing information in a Layer 3 virtual private network (VPN) environment.
- bgp.l2vpn.0 - The *bgp.l2vpn.0* routing table stores routing information in a Layer 2 VPN environment.

5.5 JUNOS software Preference Values

JUNOS 根据 preference 选择优选路由，类似于 cisco 管理距离。

Source or Protocol Name	Meaning	Preference Value
Direct	Subnet address of an interface	0
Local	Host address of a directly connected interface	0
Static	Static routes	5
RSVP	Resource Reservation Protocol	7
LDP	Label Distribution Protocol	9
OSPF Internal	Open Shortest Path First internal routes	10
IS-IS Level 1 Internal	Intermediate System to Intermediate System Level 1 internal routes	15
IS-IS Level 2 Internal	Intermediate System to Intermediate system Level 2 internal routes	18
RIP	Routing Information Protocol	100
PIM	Protocol Independent Multicast	105
Aggregate	Aggregate and generated routes	130
OSPF External	Open Shortest Path First external routes	150
IS-IS Level 1 External	Intermediate System to Intermediate System Level 1 external routes	160
IS-IS Level 2 External	Intermediate System to Intermediate System Level 2 external routes	165
BGP	Border Gateway Protocol	170
MSDP	Multicast Source Discovery Protocol	

5.6 新添加路由表

```
[edit]
routing-options {
        rib fbf-north.inet.0 {
                static {
                        route 0.0.0.0/0 next-hop 172.16.1.1;
                }
        }
        rib fbf-south.inet.0 {
                static {
                        route 0.0.0.0/0 next-hop 172.16.2.1;
                }
        }
        static {
                route 0.0.0.0/0 next-hop 192.168.0.1;
        }
}
```

- **Rib-groups**

把默认路由表 inet.0 和 inet.2 导入到 ospf-rib 在导入 ospf 中。

Example:

```
routing-options {
    rib-groups {
                ospf-rib {
                        import-rib [ inet.0 inet.2 ];
                }
        }
}
protocols {
        ospf {
                rib-group ospf-rib;
        }
}
```

5.7 Load Balancing

JUNOS 能使用策略实现 Load Balancing

Example:

```
routing-options {
        forwarding-table {
                export load-balance;
```

```
        }
}
 policy-options {
        policy-statement load-balance {
                then {
                        load-balance per-packet;
                        accept;
                }
        }
}
```

Chardonnay
10.222.10.2
10.222.20.2
10.222.10.1
10.222.20.1
Merlot
Cabernet
192.168.80.1
10.20.20.2
10.10.10.2
10.20.20.1
10.10.10.1
Riesling

(1)查看路由表

```
user@Merlot> show route 192.168.80/24 terse

inet.0: 4 destinations, 4 routes (4 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

A Destination          P Prf Metric 1   Metric 2    Next hop         AS path
* 192.168.80.1/32      I  18          20             10.222.10.2
                                                    >10.20.20.1
```

(2)查看转发表

```
user@Merlot> show route forwarding-table matching 192.168.80/24

Routing table:: inet
Internet:
Destination          Type RtRef Nexthop          Type Index NhRef Netif
192.168.80.1/32      user      0 10.20.20.0       ucst     26     30so-0/0/3.0
```

（3）使用策略路由

```
[edit]
user@Riesling# show
policy-options {
    policy-statement please-load-balance-traffic {
        then {
            load-balance per-packet;
```

```
        }
    }
}
```

转发类型可以使用下列参数：

Layer 3：

- Incoming interface
- Source IP address
- Destination IP address

基于流

- Protocol (TCP or UDP)
- Source port number
- Destination port number

（4）使用转发表

```
user@Merlot> show route forwarding-table matching 192.168.80/24

Routing table:: inet
Internet:
Destination        Type RtRef Nexthop       Type Index NhRef Netif
192.168.80.1/32    user     0                ulst    30     14
                              10.222.10.0   ucst    20     19  so-0/0/0.0
                              10.20.20.0    ucst    26     22  so-0/0/3.0
```

（5）Internet Processor ASICs and Load Balancing 关系

Internet Processor ASIC forwards packets across 8 equal-cost next hops, Internet Processor ASIC forwards packets on a per-packet basis only.

Internet Processor II ASIC, however, is able to forward across 16 equal-cost next hops. Internet Processor II ASIC is based on a microflow.

5.8 其它 Features

- Router-ID，建议使用 Lo0 的地址

```
[edit]
routing-options {
    router-id 172.168.1.1;
```

}

- Autonomous system number

配置 BGP 时候用

```
[edit]
routing-options {
    autonomous-system 65001;
}
```

6. Route policy

在大型的网络中使用控制路由选路的最有效率的方法。特别是对 BGP 控制。策略路由允许控制路由信息、路由协议影响路由表，路由表影响转发表。

6.1 policy overview

- 控制路由信息调整输入和输出路由表
  1. 忽略或改变输入的路由信息
  2. 阻止或改变输出的路由信息

- 策略用匹配条件和执行动作
  匹配条件可以是协议细节

6.2 什么时候使用路由策略

- 不愿把学习到的所有路由用于路由表。
- 不愿把学习到的所有路由发布给邻居。
- 控制路由重发布。
- 想更改路由选路。

6.3 输入输出策略

- 执行策略过滤控制路由表
  1. 在形成路由表前执行输入策略
  2. 在发布 active 路由前执行输出策略

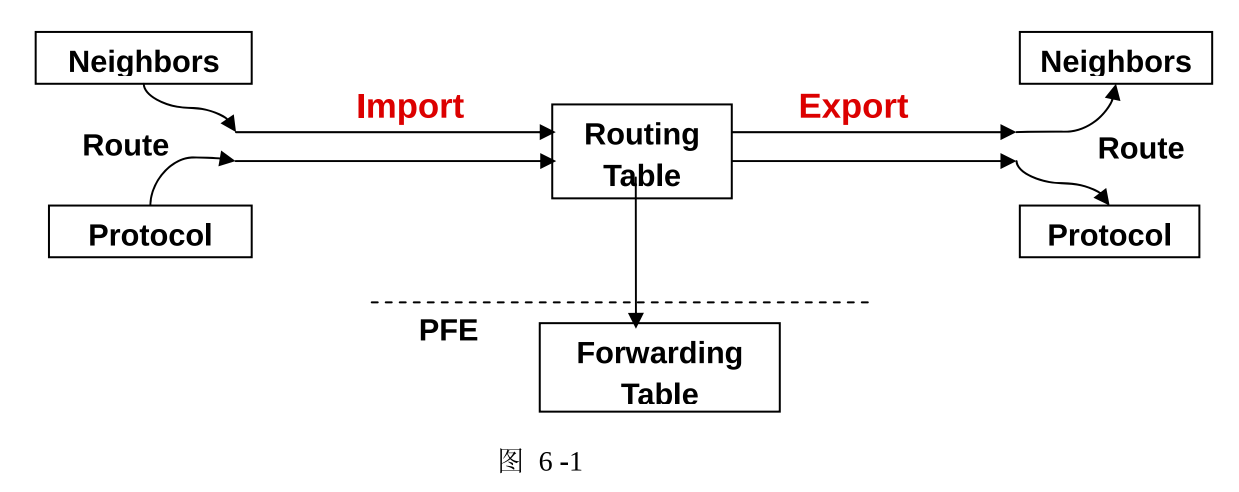


图 6-1

6.4 查看策略操作

通过 show route receive-protocol 和 show route receive-protocol 来查看执行输入输出策略路由修改的情况。

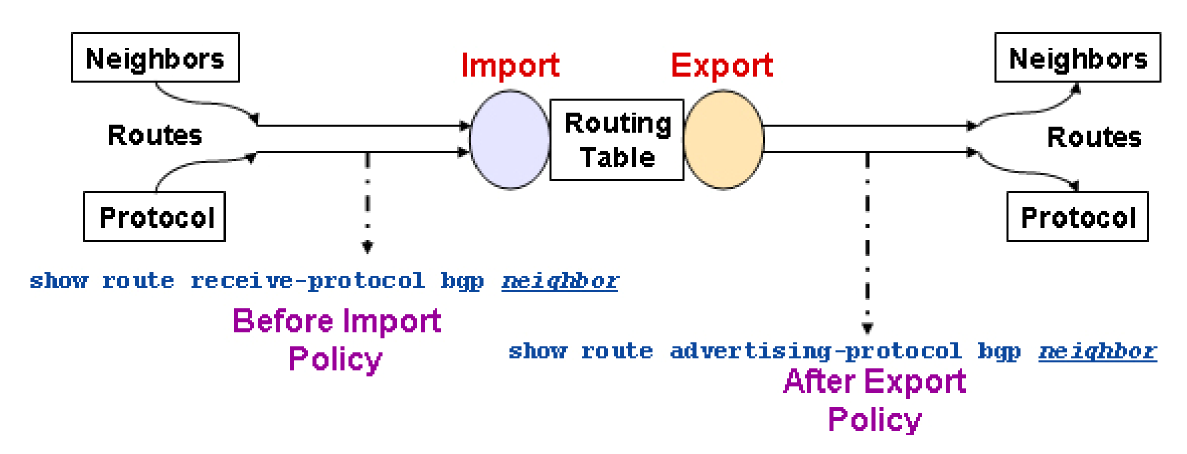


图 6 – 2

6.5 路由策略链

6.5.1 策略执行流程

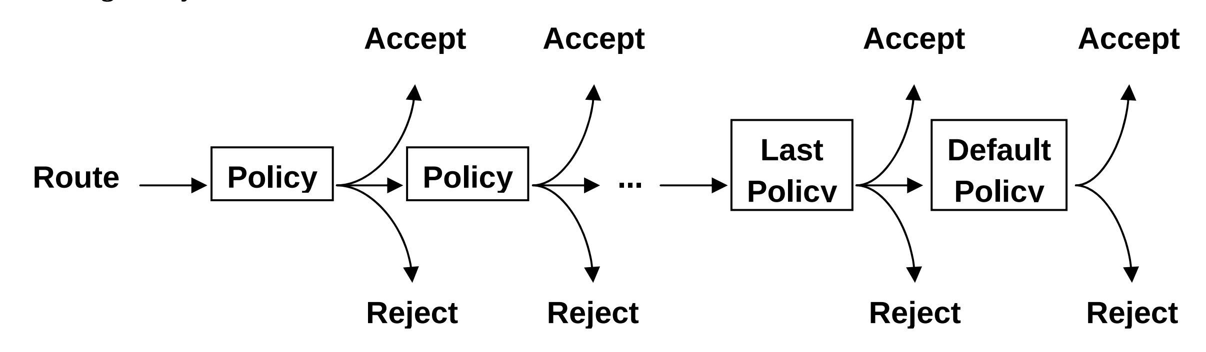
一个策略可以包含多个条件：

Syntax:

```
policy-options {
    policy-statement policy-name {
        term term-name {
            from {
                match-conditions;      /***  匹配条件  ***/
            }
            then {
                action;                /***  执行动作  ***/
            }
        }
        term term-name {
            from {
                match-conditions;
            }
            then {
                action;
            }
        }
```

    }
}

Routing Policy Flow ：



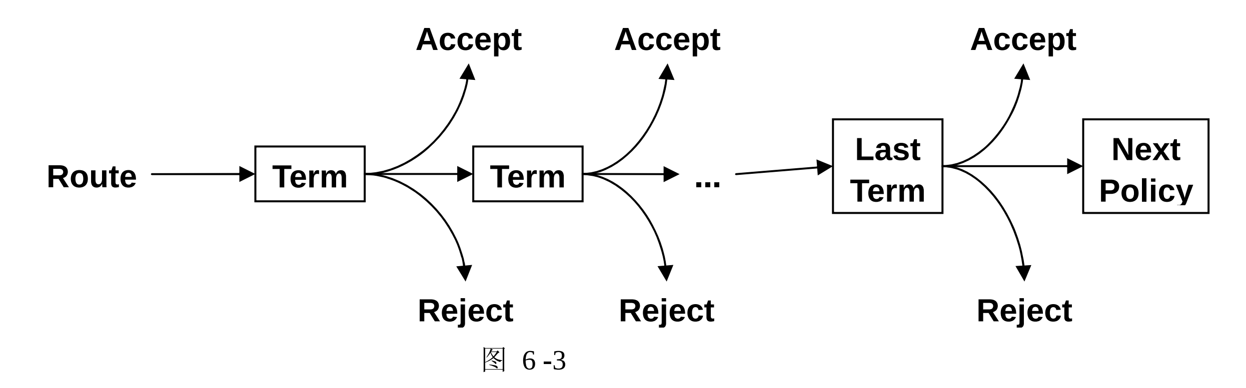


图 6 -3

6.6 匹配条件

- 每一个策略包含多个匹配条件
- 匹配条件包括：

  1. **Neighbor address**
  2. **Protocol (source of information)**
     - BGP, direct, DVMRP, IS-IS, local, MPLS, OSPF,
     - PIM, RIP, static, aggregate
  3. **Routing protocol information**
     - OSPF area ID
     - IS-IS level number
     - BGP attributes

  等

- From and To

— From 匹配的条件：area area-id、as-path name、community [names]、level level

、local-preference value、metric metric、neighbor address、next-hop address

、origin value、preference preference、protocol protocol、rib routing-table

；

to 匹配条件：level *level* 、rib *routing-table* ；

Example:

```
policy-options {
    policy-statement isis-export {
        term sending-to-neighborA {
            to level 2;
            then accept;
        }
    }
}
```

6.7 匹配后动作

- **Actions:**
  - **Terminate**
    - **Accept route**
    - **Reject (or suppress) route**
  - **Flow control**
    - **Skip to next policy**
    - **Skip to next term**
  - **Modify attributes**
    - **Metric**
    - **Preference**
    - **Color**
    - **Next-hop address**

6.8 default policy

- 在没有任何匹配情况下每个协议使用默认策略执行。
- IS—IS ， OSPF，and RIP
  - **Import all routes**
  - **Export all routes learned by that protocol and all interface routes on which the protocol is configured explicitly (except RIP)**
- BGP
  - **Import all routes learned from BGP neighbors**
  - **Export all active routes learned from BGP neighbors to all BGP neighbors**
    - — **EBGP-learned routes are exported to all BGP peers**
    - — **IBGP-learned routes are exported to all EBGP peers (logical full-mesh)**

6.9 Route Filters

- 类似 cisco ip prifex 自由的来匹配路由。

Syntax：route-filter *prefix/prefix-length match-type actions;*

参考下列模型：

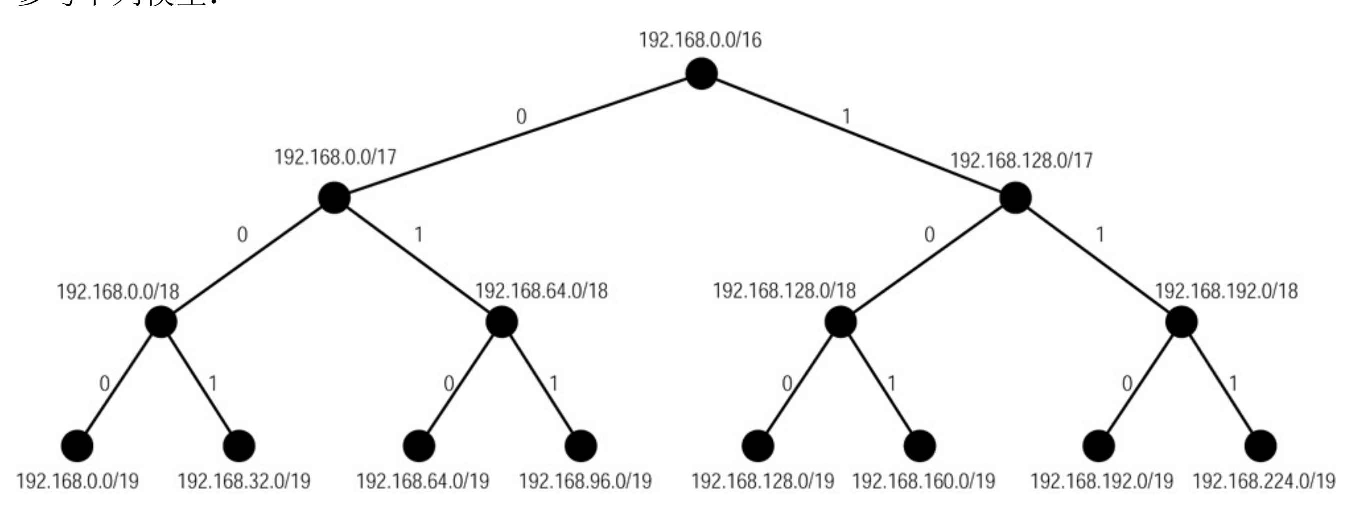


图 6 - 4

— **Exact**

route-filter 192.168.0.0/16 exact;

— **orlonger;**

route-filter 192.168.0.0/16 orlonger;

— **longer**

route-filter 192.168.0.0/16 longer;

— **upto**

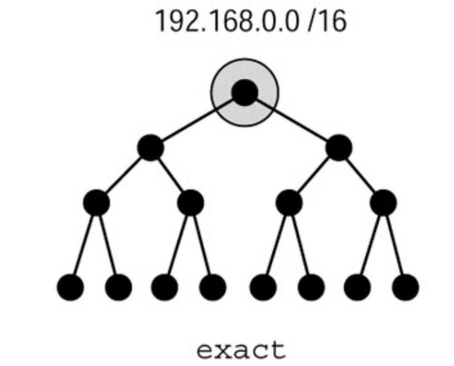
route-filter 192.168.0.0/16 upto /18;

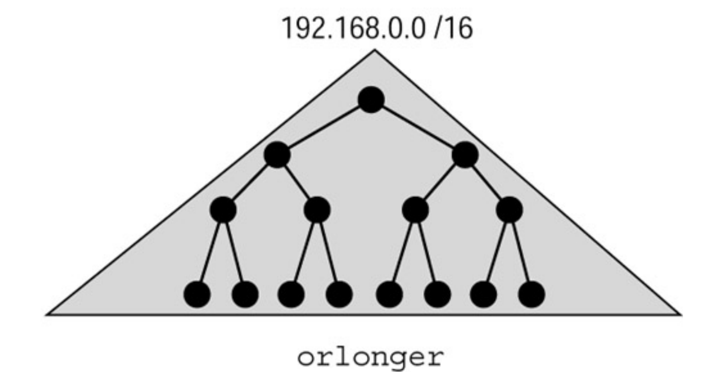
— **prefix-length-range**

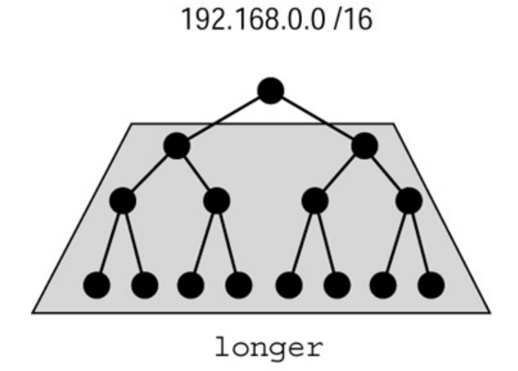
route-filter 192.168.0.0/16 prefix-length-range /17-/18;

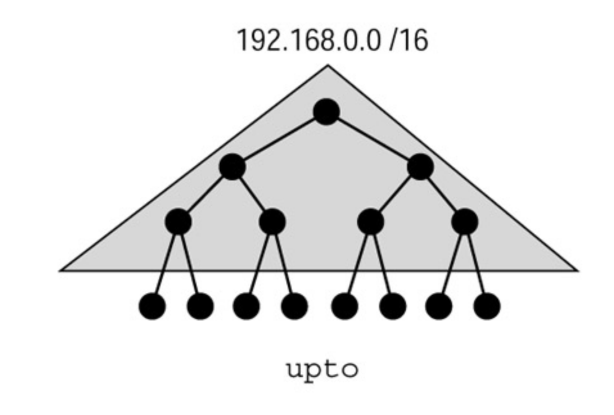
— **through**

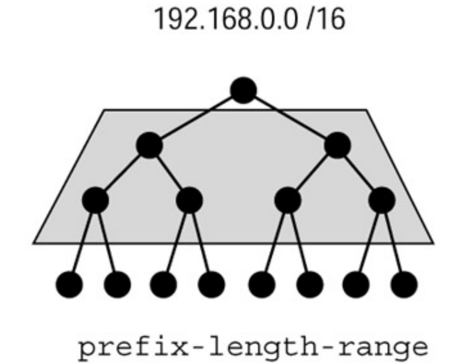
route-filter 192.168.0.0/16 through 192.168.128.0/19;











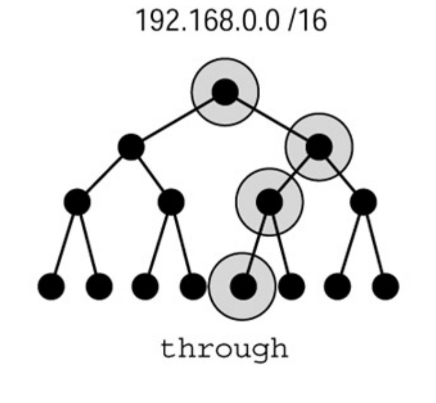


图 6 - 5

6.10 Route Filters and Other Match Criteria

首先匹配 Route Filters，Route Filter，然后匹配其他条件。
如下图所示：

```
policy-options {
    policy-statement bgp-export {
        term coming-from-neighborA {
            from {
                protocol bgp;
                metric 10;
                route-filter 192.0.0.0/8 exact;
                route-filter 192.168.0.0/16 longer;
                route-filter 10.222.12.0/24 longer;
            }
            then accept;
        }
    }
}
```


图 6－6

6.11 Applying Routing Policies

- RIP Policy Application

```
protocols {
    rip {
        import [ policy1 policy2 ...];
        group test {
            export [ policy1 policy2 ...];
            neighbor fe-0/0/2.0 {
                import [ policy1 policy2 ...];
            }
        }
    }
}
```

- Link-State IGP Policy Application

```
protocols {
    isis {
        export [ policy1 policy2 ...];
    }
    ospf {
        export [ policy1 policy2 ... ];
    }
}
```

6.12 BGP Policy Application

分为三个层次的应用策略控制。

```
protocols {
    bgp {
        import [ policy1 policy2 ...];
        export [ policy1 policy2 ...];
        group external-peers {
            type external;
            import [ policy1 policy2 ...];
            export [ policy1 policy2 ...];
            peer-as 65521;
            neighbor 1.1.1.1 {
                import [ policy1 policy2 ...];
                export [ policy1 policy2 ...];
            }
        }
    }
}
```

7. Routing Information Protocol （RIP）

7.1 RIP 技术概述

ＲＩＰ用两种分组传输信息，更新（ＵＮＰＤATＥＳ)和请求(ＲＥＱＵＥＳＴＳ)。每个有ＲＩＰ功能的路由器每隔３０秒用 UDP 520 端口给与之直接相连的机器广播更新信息。更新信息反映了该路由器所有的路由选择信息数据库。路由选择信息数据库的每个条目由两部分组成：局域网上能达到的ＩＰ地址和与该网络的距离。请求信息用于寻找网络上能发出ＲＩＰ报文的其他设备。

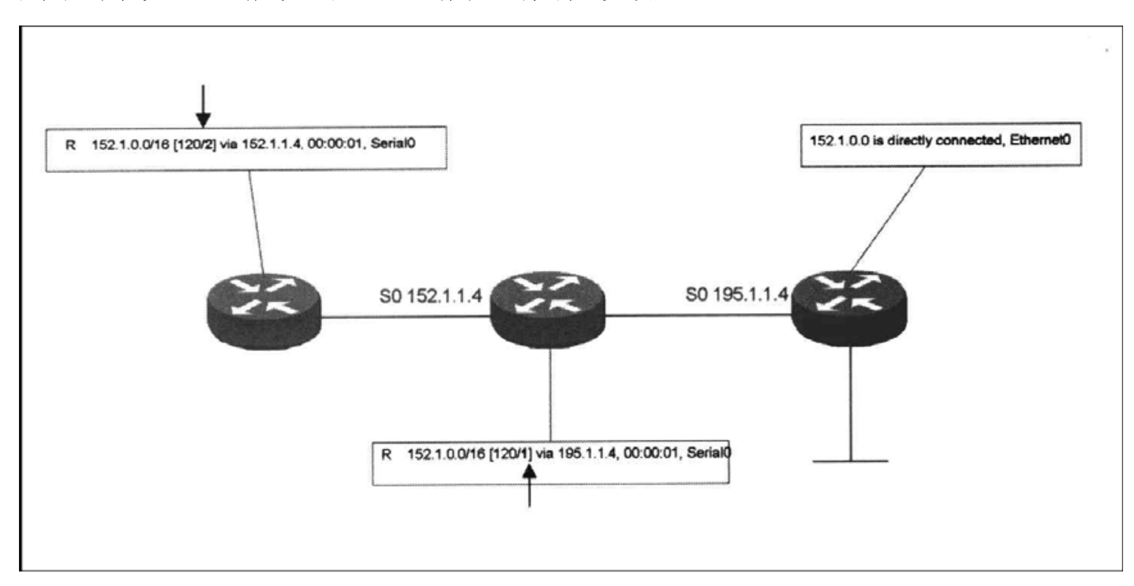


图7-1 RIP 度量

ＲＩＰ用路程段数作为网络距离的尺度。每个路由器在给相邻路由器发出路由信息时，给每个路径加上内部距离（１）。在图８－１中，路由器Ｃ直接和网络Ｃ相连。当它向路由器Ｂ通告网络１５２．１．０．０的路径时，它把度量增加１。与之相似，路由器Ｂ把度量增加到二且通告路径给路由器Ａ。路由器Ｂ和路由器Ａ与１５２．１．０．０的距离分别是１跳、２跳。如图８－１所示，到达目的地的跳数是数据报到达目的地网络必须通过的路由器数。然而用跳数作为确定路径的标准不是总能提供最短路径。例如，在图８- ２中，从路由器Ａ到网络Ｂ，ＲＩＰ将更倾向于５６ KBＰＳ链路而不是１．５Ｍｂｐｓ链路。跳数为１的５６ KBＰＳ串行链路比跳数为２的１．５Ｍｂｐｓ串行链路慢很多。

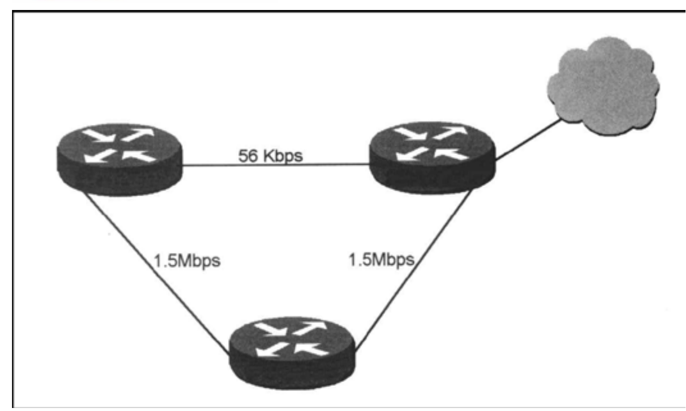


图 7-2 跳数

7.2.1 路由选择环路

任何距离向量路由选择协议如ＲＩＰ，都有一个问题，路由器不知道网络的全局情况。路由
器必须依靠相邻路由器来获取网络的可达信息。由于路由选择更新信息在网络上传播慢，距
离向量路由选择算法有一个慢收敛问题，这个问题将导致不一致性产生。ＲＩＰ使用以下机制减
少因网络上的不一致带来的路由选择环路的可能性：
记数到无穷大、水平分割、破坏逆转更新、保持计数器和触发更新。
1. 记数到无穷大问题
ＲＩＰ允许最大跳数为１５。大于１５的目的地被认为是不可达。这个数字限制了网络大小的同
时也防止了一个叫做记数到无穷大的问
题，如图８－３所示。
1) 记数到无穷大如下工作：路由器 A
丢失了以太网接口后产生一个触发更新送
往路由器 B 和路由器 C。这个更新信息告
诉路由器 B 和路由器 C 路由器 A 不再到达网
络 A 的路径。这个更新信息传输到路由器
B 被推迟了（ＣＰＵ忙、链路拥塞等）但到
达了路由器 C。路由器 C 从路由表中去掉
到网络 A 的路径。
2) 路由器 B 仍未收到路由器 A 的触发

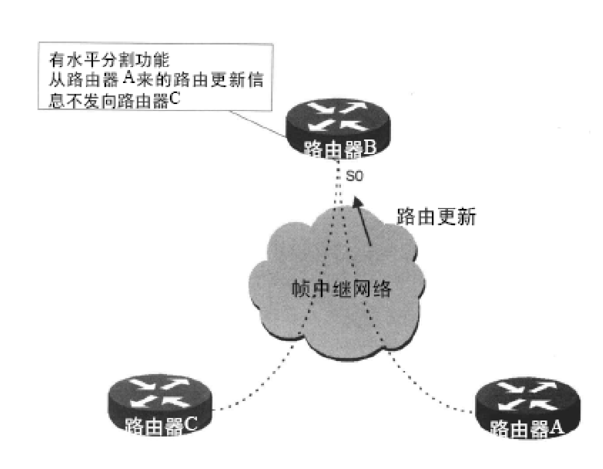


图 7-3 计数无穷大

更新信息，并发出它的常规路由选择更新信息，通告网络 A 以 2 跳的距离可达。路由器 C 收到
这个更新信息，认为出现了一条新路径到网络 A。
3) 路由器 C 告诉路由器 A 它能以 3 跳的距离到达网络 A。
4) 路由器 A 告诉路由器 B 它能以 4 跳的距离到达网络 A。

5）这个循环将进行到跳数为无穷，在ＲＩＰ中定义为１6。一旦一个路由器达到无穷，它将声明这条路径不可用并从路由表中删除此路径。
由于记数到无穷大问题，路由选择信息将从一个路由器传到另一个路由器，每次段数加
一。路由选择环路问题将无限制地进行下去，除非达到某个限制。这个限制就是ＲＩＰ的最大跳
数。当路径的跳数超过１５，这条路径就从路由表中删除。

2. 水平分割
水平分割规则如下：路由器不向路径到来的方向回传此路径。当打开路由器接口后，路由器记录路径是从哪个接口来的，并且不向此接口回传此路径。Ｃｉｓｃｏ可以对每个接口关闭水平分割功能。这个特点在 non broadcast mutilple access（ＮＢＭＡ）非广播多路访问 hub-and-spoke 环境下十分有用。在图６－４中，路由器Ｂ通过帧中继连接路由器Ａ和路由器Ｃ，两个ＰＶＣ都在路由器Ｂ的同一个物理接口中止。
在图８－４中，如果在路由器Ｂ的水平分割未被关闭，那么路由器Ｃ将收不到路由器Ａ的路由选择信息（反之亦然）。用 no ip split-horizon 接口子命令关闭水平分割功能。
3. 破坏逆转
水平分割是路由器用来防止把一个接口得来的路径又从此接口传回导致的问题的方案。
水平分割方案忽略在更新过程中从一个路由器获取的路径又传回该路由器。有破坏逆转的水平分割的更新信息中包括这些路径，但这个处理过程把这些路径的度量设为１６（无穷）。
通过把跳数设为无穷并把这条路径告诉源路由器，有可能立刻解决路由选择环路。否则，不正确的路径将在路由表中驻留到超时为止。
破坏逆转的缺点是它增加了路由更新的的数据大小。

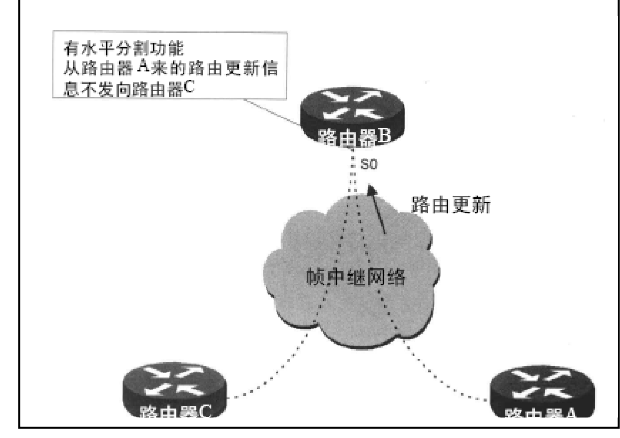


图 7-4　水平分割

4. 保持
保持定时器防止路由器在路径从路由表中删除后一定的时间内接受新的路由信息。它的思想是保证每个路由器都收到了路径不可达信息，而且没有路由器发出无效路径信息。例如在图８－３中，由于路由更新信息被延迟，路由器Ｂ向路由器Ｃ发出错误信息。使用保持计数器这种情况将不会发生，因为路由器Ｃ将在１８０秒内不接受通向网络Ａ的新的路径信息。到那时路由器Ｂ将存储正确的路由信息。
5. 触发更新
有破坏逆转的水平分割将任何两个路由器构成的环路打破。三个或更多个路由器构成的环路仍会发生，直到无穷（１６）时为止。触发式更新想加速收敛时间。当某个路径的度量改变了，路由器立即发出更新信息，路由器不管是否到达常规信息更新时间都发出更新信息。

7.2.2 RIP 报文格式

如图８－５所示为ＲＩＰ信息格式。３２位信息头后是一系列的对子。这些对子包括网络的ＩＰ地址和一个用于表示到达网络的距离。Commands (命令): 一般情况下，命令要么是ＲＩＰ请求（1），要么是ＲＩＰ更新（ 2）。命令３和４已经废弃不用，命令５被 Sun Microsystems 保留作内部使用。Ｖｅｒｓｉｏｎ（版本）：这个域包含协议版本号。目前有两种ＲＩＰ版本。Address Family Identifier（地址族标识符）：ＲＩＰ能给多个协议传输路由信息。这个域用于指定被传送协议的族。ＩＰ的地址族标识是 2。IP Address（ＩＰ地址）：这个域包含ＩＰ地址，ＩＰ地址长为４个字节。Must Be Zero（必须是零）：ＲＩＰ能传输高达１２个字节。由于ＩＰ地址用了１２个字节中的4个字节，余下的8个字节都填零。Distance to Net（到网络的距离） ：这个域包含一个用于描述到指定网络距离的整数。若网络不可达，则此值为１６。

0	8	16	24	31
命令 (1-5)	版本 (1)	必须为0		
网络1的地址族		必须为0		
网络1的IP地址				
必须为0				
必须为0				
网络1的距离				
网络2的地址族		必须为0		
网络2的IP地址				
必须为0				
必须为0				
网络2的距离				

图 7-5 RIP信息格式

详细了解可以参阅 RFC

- RFC 1058, "Routing Information Protocol"
- RFC 2082, "RIP-2 MD5 Authentication"
- RFC 2453, "RIP Version 2"

7.3 RIP 配置

— 最小 RIP 配置：

```
 [edit protocol]
protocols {
    rip {
        group group-name {
            neighbor interface-name;
        }
    }
}
```

— RIP 重发布和 Export 策略

```
policy-options {
    policy-statement statics-to-rip {
        from protocol static;
        then accept;
    }
}
```

— 应用策略的邻居

```
protocols {
    rip {
        group rip-neighbors {
            export statics-to-rip;
            neighbor fe-0/0/0.0;
            neighbor fe-0/0/1.0;
        }
    }
}
```

7.3 Monitoring RIP

7.3.1 查看 RIP 邻居关系

```
user@Cabernet> show rip neighbor
                          Source      Destination     Send    Receive   In
Neighbor         State    Address     Address         Mode    Mode      Met
---------        -----    -------     -----------     ----    ----      ---
```

```
fe-0/0/0.0     Up      172.16.1.2      224.0.0.9       mcast    both         1
fe-0/0/1.0     Up      172.16.2.1      224.0.0.9       mcast    both         1
```

7.3.2 查看 RIP 路由表

```
user@Riesling> show route protocol rip

inet.0: 27 destinations, 27 routes (27 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.16.2.0/24       *[RIP/100] 00:07:25, metric 2
                     > to 172.16.1.2 via fe-0/0/0.0
192.168.8.1/32      *[RIP/100] 00:07:25, metric 2
                     > to 172.16.1.2 via fe-0/0/0.0
192.168.24.1/32     *[RIP/100] 00:00:25, metric 3
                     > to 172.16.1.2 via fe-0/0/0.0
```

7.3.3 查看 RIP 发布到指定邻居的路由

lab@sf-pe> **show route advertising-protocol rip 10.0.21.1**

```
inet.0: 12 destinations, 12 routes (11 active, 1 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

3.0.0.0/8          *[Static/5] 00:10:56
                       Reject
30.0.0.0/16        *[Static/5] 00:12:20
                       Reject
```

7.3.4 查看 RIP 收到指定邻居的路由

lab@p1> **show route receive-protocol rip 10.100.3.2**

```
inet.0: 17 destinations, 18 routes (17 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both

172.20.4.0/24      *[RIP/100] 00:01:01, metric 2
                      > to 10.100.3.2 via fe-0/0/0.0
192.168.28.0/24    *[RIP/100] 00:01:01, metric 2
                      > to 10.100.3.2 via so-0/0/0.0
```

7.3.5 查看 RIP 的 statistics

lab@sf-pe> **show rip statistics**

```
RIP info:    port 520; update interval 30s; holddown 180s; timeout 120s.
        rts learned    rts held down    rqsts dropped    resps dropped
                    1                1                0                0

fe-0/0/0.0:    1 routes learned; 3 routes advertised
Counter                              Total      Last 5 min    Last minute
-------                          -----------   -----------   -----------
Updates Sent                           28            11              2
Triggered Updates Sent                  1             0              0
Responses Sent                          0             0              0
Bad Messages                            0             0              0
RIPv1 Updates Received                  0             0              0
RIPv1 Bad Route Entries                 0             0              0
RIPv1 Updates Ignored                   0             0              0
RIPv2 Updates Received                 14            11              3
RIPv2 Bad Route Entries                 0             0              0
RIPv2 Updates Ignored                   0             0              0
Authentication Failures                 0             0              0
RIP Requests Received                   0             0              0
RIP Requests Ignored                    0             0              0
```

7.3.5 存放 RIP log

— Log file-related configuration options

```
[edit protocols rip]
    traceoptions {
        file name <replace> <size size> <files number> <no-stamp>
      }
[edit protocols rip]
lab@p1# show traceoptions file ?
Possible completions:
  <[Enter]>              Execute this command
  files                  Maximum number of trace files (2..1000)
  no-stamp               Don't timestamp trace file
  no-world-readable      Don't allow any user to read the log file
```

replace   Replace trace file rather than appending to it
size   Maximum trace file size (10240..4294967295)
world-readable   Allow any user to read the log file
| Pipe through a command

— **RIP trace flag options**

```
[edit protocols rip]
traceoptions {
    flag flag flag-modifier disable;
}
```

all   Trace everything
auth Trace RIP authentication
errorTrace RIP errors
expiration   Trace RIP route expiration processing
general   Trace general events
holddown   Trace RIP holddown processing
normal   Trace normal events
packets   Trace all RIP packets
policy   Trace policy processing
request   Trace RIP information packets
route   Trace routing information
state Trace state transitions
task Trace routing protocol task processing
timer   Trace routing protocol timer processing
trigger   Trace RIP triggered updates
update   Trace RIP update pacake

7.4 配置案例

按照图 8-6 拓扑。实现 RIPv2 互联。互联地址 192.168.100.0， /30 的互联网段，loopback 地址 192.168.101.0 /32 的地址。

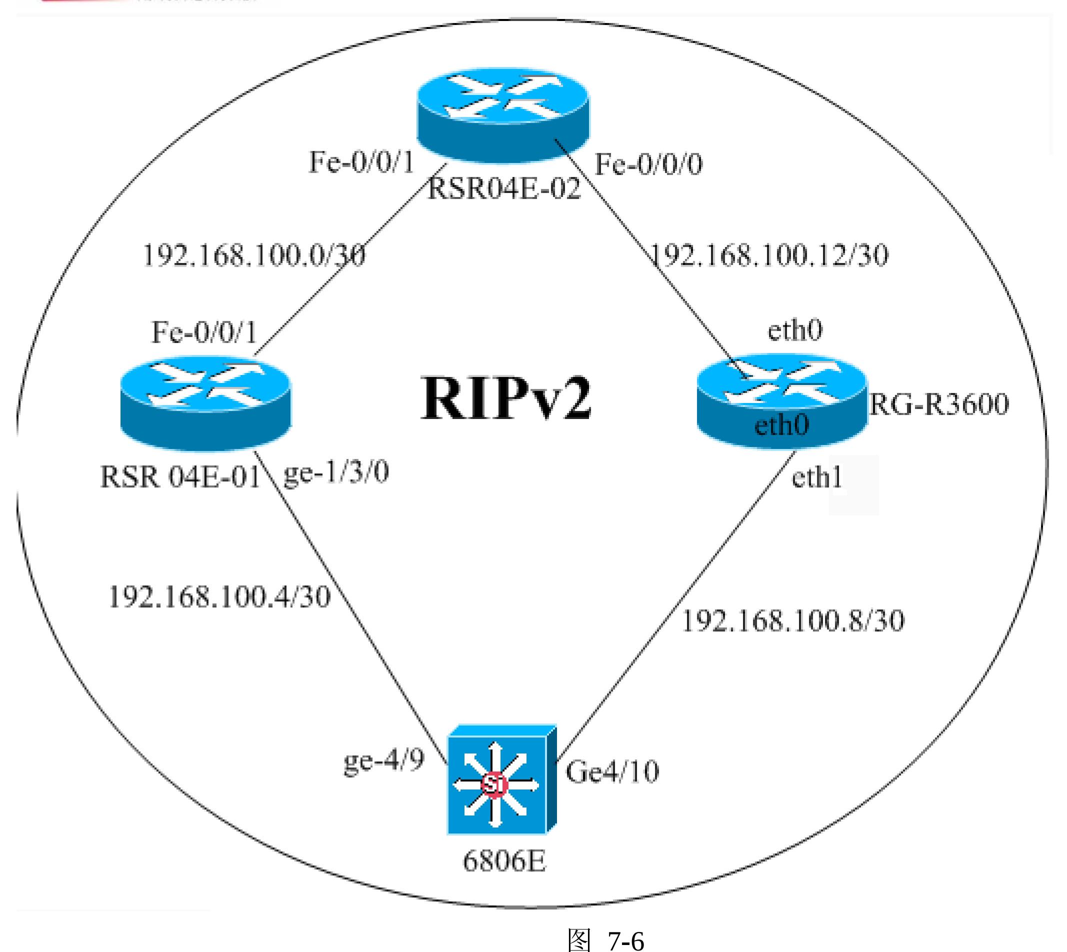


图 7-6

Example :

dmin@**RSR04E-1**# show

```
rip {
    traceoptions {
        file rip-m size 1m;
        flag all send receive detail;
    }
    group neighbor-router {
        neighbor fe-0/0/1.0;
        neighbor ge-1/3/0.0;
    }
}
```

[edit protocols]

admin@**RSR04E-2**# show

```
rip {
    traceoptions {
        file rip-m size 1m;
        flag all send receive detail;
    }
    group neighbor-router {
```

```
            neighbor fe-0/0/1.0;
            neighbor fe-0/0/0.0;
        }
}
```

admin**@RSR04E-1# run show rip neighbor**

```
                              Source           Destination     Send   Receive   In
Neighbor              State   Address          Address         Mode   Mode      Met
--------              -----   -------          -----------     ----   -------   ---
ge-1/3/0.0              Up 192.168.100.5     224.0.0.9        mcast   both        1
fe-0/0/1.0              Up 192.168.100.1     224.0.0.9        mcast   both        1
```

admin**@RSR04E-2**> **show rip neighbor**

```
                              Source           Destination     Send   Receive   In
Neighbor              State   Address          Address         Mode   Mode      Met
--------              -----   -------          -----------     ----   -------   ---
fe-0/0/0.0              Up 192.168.100.13    224.0.0.9        mcast   both        1
fe-0/0/1.0              Up 192.168.100.2     224.0.0.9        mcast   both        1
```

8. Open Shortest Path First (OSPF)

8.1 引言

开放式最短路径优先协议是一种链路状态路由选择协议，它产生于 I P 网络，发展成用于单个自治系统来分发路由选择信息。下面的内容将讨论关于该协议的术语、重要概念、配置问题和故障查找技术。当然是在使用了 O S P F 的网中。

OSPF 术语

当处理 O S P F 时，你应懂得所要使用的术语。

• 自治系统（A S)：它包括一个单独的管理实体下所控制的一组路由器。例如，属于一个特定公司的所有路由器。

• 链路状态通告（ L S A)：L S A 用来描述路由器的本地状态， L S A 包括的信息有关于路由器接口的状态和所形成的邻接状态， L S A 充满了网络。L S A 所包含的信息靠域中的每一个路由器来发送，这些域形成了路由器的拓扑数据库。到达一个目的地的最短路径，可从这些信息中算出。

• 区域：一个区域是指一个路由器的集合，它有一个一样的拓扑数据库， O S P F 用区域把一个 A S 分成多个链路状态域，因为一个区域的拓扑结构对另一个区域是不可见的，一个区域不会被扩散，这个特征大大降低了一个 A S 中的路由交通数量，区域被用来包含链路状态的更新并使管理者能建立分层网络。

• 费用（ C o s t)：是一种标准，路由器用来比较各条路径到同一目的耗费。用低的路径是最好的，O S P F 根据带宽来计算使用一种链接的费用，带宽越宽、费用越低越好。

• 路由器标识（Router ID)：路由器标识是一个 3 2 位数字，它被赋给 O S P F 使得每个路由器能够被用来独一无二地识别 A S 中的路由器，在启动时刻计算的路由器标识是路由器中的最高回送地址，如果没配备回送接口，将使用最高的 I P 地址。

• 邻接（ A d j a c e n c y)：O S P F 在相邻路由器间建立邻接，使之能交换路由信息，在多路访问网络中，每个路由器与指定的路由器形成邻接。

• 指定路由器（ D R)：它是用来降低邻接的数目，这些邻接形成于一些多路访问网络中，例如以太网、令牌环、帧中继。这些形成的邻接的数量降低大大降低拓扑数据库的规模，D R 与多路访问网中的其它所有路由器形成邻接，这些路由器把它们的 L S A 发给 D R，而 D R 的任务是把这些 L S A 发送到整个网络， D R 中的思想是路由器都有一个发送信息的汇点，同时，每个路由器与网络中其它路由器交换信息。

• 备用指定路由器（ B D R)：它应用于一个多路访问网络中，它的任务中是在 D R 失效时，接管它。

• 区域间路径(I n t e r-Area Route) ：它是产生于一个非本地区域中的路径，存在于当前的 O S P F 路由域中。

• 区域内路径(Intra-Area route)：它是一个区域内的路径。

• 邻居（ N e i g h b o r)：它是一些共享同一网络的路由器，例如在一个以太网接口上的两个路由器称为邻居。

• 扩散(F l o o d i n g)：是一种在路由器间分发 L S A 的技术。

• H e l l o：一个 H e l l o 包用于建立和维护邻居间的关系， H e l l o 包同样用来在网中选择一个 D R。

8.2 OSPF 技术概述

在进入细节以前让我们从对 O S P F 的一个简要的介绍开始，O S P F 用链路状态算法来计算在每个区域中到所有目的的最短路径，当一个路由器首先开始工作，或者任一个路由变化发生，这个配备给 O S P F 的路由器将 L S A 扩散到同一级区域内所有路由器，这些 L S A 包含这个路由器的链接状态和它与邻居路由器联系的信息，从这些 L S A 的收集中形成了链路状态数据库，在这个区域中的所有路由器都有一个特定的数据库来描述这个区域的拓扑结构。
这个路由器于是就运行 D i s k j t r a 算法，这个算法利用链路状态数据库在该区域中形成到所有目的的最短路径树，从这个最短路径树中形成了 I P 路由表。在网络中发生的任何改变将会被链路状态包扩散出去，同时使路由器利用这些新信息，重新计算最短路径树。

8.2.1 链路状态路由协议

O S P F 利用链路状态算法来计算到所有已知目的的最短路径。链路状态指的是一个路由的接口的状态（包括与上、下、I P 地址、网络类型等）和路由器和它邻居间的联系，这个路状态通告被扩散到每个路由器并用来建立一个拓扑数据库， D i s k j t r a 算法被运行在每个使了拓扑数据库的路由器上，这个数据库是靠收到区域内所有路由器发来的 L S A 而产生的，算法被放置到处于树根处的路由器上，它根据到达这个网络的费用计算到达目的的最短路径。

8.2.2 扩散

扩散是一个在相邻路由器间分发链路状态通告的过程，扩散过程使 L S A 从它的源点进行下一跳，因为在一个 O S P F 域中的所有路由器都靠邻接关系互联，所以信息可以在整个网络中传播。为了使这个过程可靠，每个链路状态通告都必须被应答。

8.2.3 Diskjtra 算法

D i s k j t r a 算法是 O S P F 的核心，一旦路由器收到所有的链路状态通告，路由器就利用 D i s k j t r a 算法来计算在这个区域中到每一个目的地的最短路径（根据到那个目的累加费用)。每一个路由器将会对区域中的网络拓扑结构有一个完整的观察，以自己为根生成一个树，并且有着到达任一个目的网络或主机的完整道路。
一个路由器对拓扑结构的观察将不同于其它的路由器，这是因为每个路由器把它自己作为树的根，任何时候运行 D i s k j t r a 算法都会获得一个新的链路状态通告。

8.2.4 区域

O S P F 利用区域来给 A S 分段，并包括链路状态更新，如图 9 - 1 所示， L S A 只被扩散到一个区域中，所分割一个区域将会降低网络中的交通数量。
在一个区域中的路由器都有一个特定的拓扑数据库，就像同区域中的其它路由器一样。
一个多区域中的路由器有着不同的拓扑数据库，用于不同的区域，它们都与路由器相连。那些所有的接口都在同一个区域中的路由器称为内部路由器（ I R），连接于同一自治系统中的路由器称为区域边界路由器(A B R)，另一种路由器充当网关的作用，从一个 A S 到另一个 A S 重分配路由信息，称为自治系统边界路由器(A S B R)。

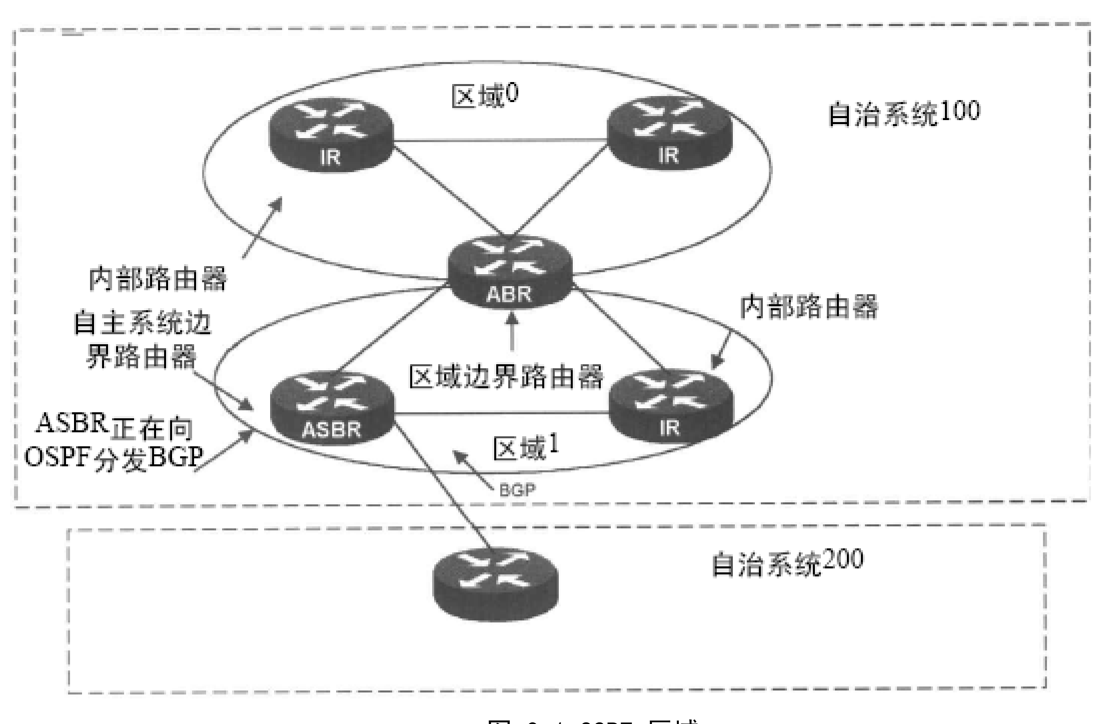


图 8-1 OSPF 区域

8.2.5 主干区域 0

O S P F 有一个主干区域的概念，指的是区域 0，如果配备了多个区域，其中一个区域必须被配备为区域 0，主干（区域 0）是所有区域的中心（所有的区域都必须与主干相连），如果一个区域没有与主干形成直接的物理连接，就必须建立一个虚链接，本章以后的内容将讨论虚链接。所有的区域向主干区域输入路由信息，并且主干也将路由信息传播回每个区域。

8.2.6 指定路由器

所有的多路访问网络都有两个或更多的附属路由器，它们会选出一个 D R。D R 的设计思想是使邻接的数目减少，这些邻接是网络中必要的。为了让 O S P F 使路由器能够交换路由信息，

必须要有一个邻接。如果没有使用 D R，在一个多路访问网络中的每一个路由器需要形成一个与其它路由器的邻接（因为链路状态数据库在邻接中是同时产生的），这个过程将会产生 N - 1 个邻接。然后，在一个多路访问网络中的所有路由器，只向 D R 和 B D R 发送路由信息，同时 D R 负责把这些信息扩散到所有邻接路由器去，并以网络的名义产生一个网络链接通告。如果 D R 失效，备份 D R 将代替它工作。邻接的减少也减少了路由协议交通的容量，以及拓扑数据库的大小。D R 使用 H e l l o 协议，这个协议在前文讨论过，优先级高的路由器决定 D R 的选择，它们被装载在 H e l l o 包中。优先级最高的路由器将被选为 D R。路由器标识是一个回送接口的 I P 地址，如没有配置回送设备，路由器标识是路由器上最高位 I P 地址。用 ip ospf priority 命令可以将路由器优先级配备给路由器接口。当一个路由器第一次在一个多路访问网中工作时，它会检查看网络中当前是否有一个 D R，如果有，就接收它而不管优先级，一旦一个 D R 被选出，其它路由器是无法代替的，除非这个 D R 失效了。如果网络中没有 D R，则路由器根据优先级谈判决定 D R。

8.2.7 OSPF 协议包

O S P F 协议直接在 I P 8 9 协议上运行，它以一个 2 4 字节的头开始(见图 8 - 2)。

<table>
<tr><td>0</td><td>8</td><td>16</td><td>24</td><td>31</td></tr>
<tr><td>版本号</td><td>OSPF包类型</td><td colspan="3">包长度</td></tr>
<tr><td colspan="5">路由器标识</td></tr>
<tr><td colspan="5">区域标识</td></tr>
<tr><td colspan="2">校验和</td><td colspan="3">确认类型</td></tr>
<tr><td colspan="5">确认</td></tr>
<tr><td colspan="5">确认</td></tr>
</table>

图 9-2 OSPF 包

有 5 种 O S P F 包类型(见图 9- 3)。

类型	包名称	协议功能
1	Hello	发现并维持邻居
2	数据库描述	概括数据库容量
3	链路状态请求	请求数据库信息
4	链路状态更新	数据库更新
5	链路状态应答	应答

图 9-3 OSPF 包的类型

1) Hello 包：H e l l o 协议的责任是发现邻居并维持邻居关系。H e l l o 包被周期地发向路由器接口，以网络类型为根据。H e l l o 协议还担负着在多路访问网络中挑选出 D R。D R 的地位在本章前文中已讨论过。

2) 数据库描述包：数据库描述包是 O S P F 的第二类包，这类包的任务是描述路由器的链路状态数据库的容量，并且是形成邻接的第一步，数据库描述器包通过一个投票响应方式发出，一个路由器被指定为主机，其它的被指定为从机，主机发出数据库选票，从机通过发出数据库描述器包来发出应答。

3) 链路状态请求包：链路状态请求包是 O S P F 的第三类包，一旦整个数据库使用数据库描述包来与路由器交换，路由器将比较它邻居的数据库和它自己的。此时，路由器也许会发现邻居的数据库在某些部分比自己的更先进。如果这样，路由器将会要求这部分使用链路状态请求包。

4) 链路状态更新包：链路状态更新包是 O S P F 包的第四类包，路由器使用扩散技术来传递 L S A，L S A 有很多类（路由器、网络、概括和外部）。

5) 链路状态应答包：链路状态应答是 O S P F 的第四类包，它用来收到 L S A 时进行应答，这种应答使 O S P F 的扩散过程更可靠。

8.2.8 链路状态通告

在图 8 - 1 中所谈到的每种路由器都生成一种不同的链路状态通告，虽然 L S A 的种类很多，我们只讨论 4 种主要的 L S A。

所有的链路状态通告以同样的 O S P F 的 2 0 个字节的头开始（见图 9 - 4）

0	16	24	31
链路状态时间	选项	链路状态种类	
链路状态标识符			
通告路由器			
链路状态序数			
链路状态校验码	长度		

图9-4 链路状态

链路状态时间：以秒计算的从链路状态产生的时间。选项：路由器提供的可供选择的功能。链路状态类型：链路状态通告的种类。链路状态标识符：这个域用来识别因特网环境的分配，这是靠通告来描述的。通告路由器：它是路由器的标识符，是从包中产生的。链路状态序数：用来检测老的或复制的链路状态通告。链路状态校验和：是用来校验链路状态通告的容量。

长度：链路状态通告的字节的长度，包括2 0个字节的头。

1. 路由器链路

在一个区域中的每个路由器产生一个路由器L S A（第1类的L S A），这个通告描述了路由器接口到这个区域的状态和费用，所有到那个区域的路由器链接都会被描述成一个单独的路由L S A。这个路由器L S A只在这个单独的区域中扩散。

2. 网络链路

网络链路通告是第2类L S A，每个多路访问网络的D R都有不少于一个的附属路由器，它产生一个网络通告，这个通告描述了网中的所有附属路由器，包括D R本身。

3. 汇总链路

汇总链路是第3类和第4类L S A，A B R产生汇总L S A，它描述了到一个单独目的路径，汇总L S A只在一个单独的区域中发布，并且所表述的目的在区域的外部，但它仍是同一个A S中的部分。主干里只发布区域内的路径。

4. 外部链路

A S B R产生一个外部的第5类L S A，它发布路由器知道的目标，它是A S外部。A S外部第5类L S A被用来发布到A S内的缺省路径。

有2类外部路径，外部种类1和外部种类2，这两类的不同之处在于，路径计算成本与度量的方法不同，外部种类1路由器使用外部成本加上内部成本，来计算一个路径，种类2比种类1更受欢迎，并且它被认为是缺省类在一条路径被重新载入O S P F中时。

5. 如何工作

当一个装备了O S P F的路由器工作时，路由器发一个H e l l o包给一个组播地址2 2 4 . 0 . 0 . 5，于是这个包被周期性地发送所有配备了O S P F的接口上，这取决于接口类型。对于广播媒介如以太网、令牌环或点到点接口， H e l l o包是每1 0秒发送一次，在N B M A上如帧中继或AT M，H e l l o包每3 0秒发送一次。

H e l l o包不仅被用来建立邻居连系，并发现哪个邻居在同一条线上，它还被用来描述路由器的那些可供选择的功能，如路由器是否处在一个常规或残余区域中， H e l l o包还被用来挑选出多路访问网络中的D R。在邻居被发现后，双向通信确立，指定路由器被挑出（在一个多路访问媒介上），路由器就设法与邻居路由器形成一个邻接。为了形成一个邻接，路由器必须使它们的数据库同步。为了完成这个任务，每个路由器通过发送一个连续的数据库描述包而向其它路由器描述它的数据库。这个过程被称为数据库交换过程并会在本章后文中详细涉及。

在数据库交换过程中，两路由器形成一个主从关系，主路由器发送的每个数据库描述包包含一个序数。从路由器通过目送个序数来应答表示接收。在数据库交换过程中，每个路由器检查它的数据库看它的邻居所受到的链路状态通告是否比它自己的数据库复本更新，路由器对此作记录，并且在数据库交换过程结束后，路由器要求更新L S A通过使用一个链路状态请求包。每个路由器用一个链路状态更新去响应那个链路状态请求，当请求的路由器收到了一个更新的L S A，它响应这个包，当数据库描述过程结束了并且所有链路状态请求被更新，数据库被同步了。

6. 邻接如何形成

为了使一个路由器能与另一个路由器交换链路状态数据库信息，需建立一个邻接。这个过程是O S P F的一个重要部分。在一个RSR的路由器上，可以通过使用show ospf neighbor命令来检查邻接的状态。下面的例子将告诉你命令的输出。注意，邻接的状态是满的，这意味着路由器B的数据库已与邻接的10.0.1.1路由器A的同步了。

```
user@Chardonnay> show ospf neighbor
  Address          Interface            State      ID                Pri
Dead
10.0.1.46          at-0/1/0.100         Full       10.0.1.103        128    36
10.0.1.1           so-0/0/3.0           Full       10.0.1.23         128    39
```

在完整形成一个邻接或产生一个完整邻居状态前邻居路由器要经历5个状态：

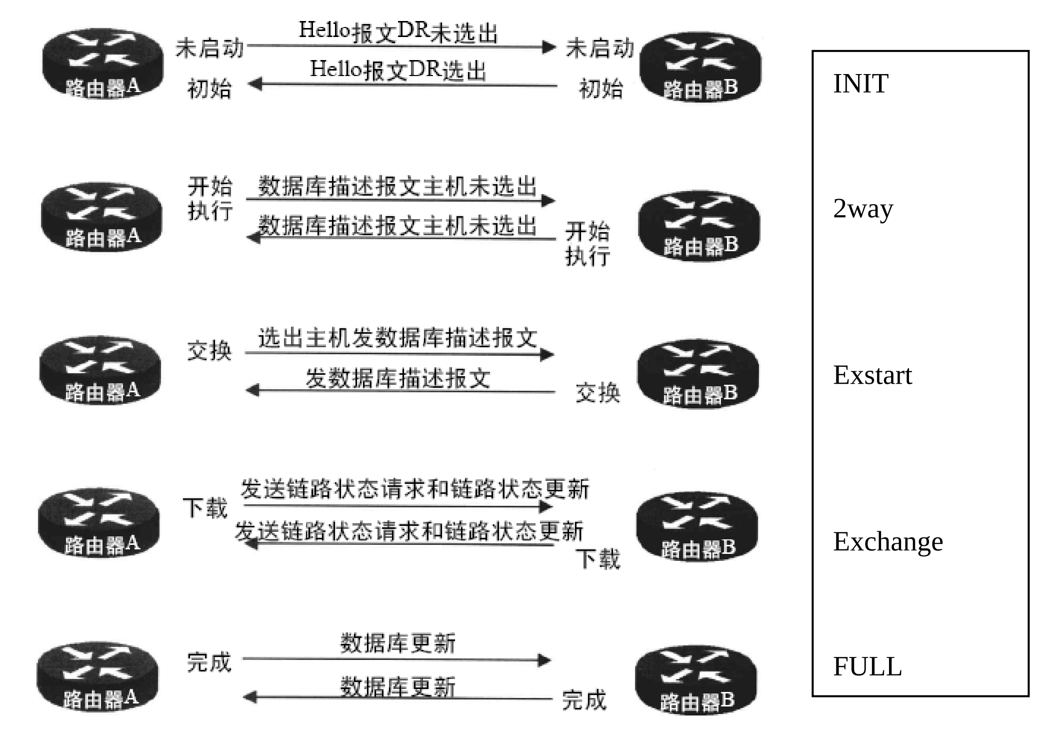


图 8-5 一个路由器如何形成邻接

8.2.9 OSPF 网的类型

OSPF有4种网络类型或模型（广播式、非广播式、点到点和点到多点），根据网的类型不同，OSPF工作方式也不同，懂得OSPF在各种网络模型上如何工作很重要，特别是在设计一个稳定的强有力的网络时。

1. 广播式

广播式网络类型是LAN上的缺省类型。

2. 非广播式

非广播式网络是串行接口上的缺省类型。

3. 点到点式

点到点的网络类型是串行口的缺省类型。

4. 点到多点

配置在NBMA下。

8.3 配置 OSPF

8.3.1 单区域

[edit]
user@host# **set protocols ospf area 0 interface ge-0/0/0**

```
[edit]
user@host# show protocols ospf
ospf {
        area 0.0.0.0 {
                interface ge-0/0/0.0;
        }
}
```

8.3.2 配置 OSPF 多区域

```
[edit]
user@host# show protocols ospf
ospf {
        area 0.0.0.0 {
                interface ge-0/0/0.0;
        }
}

[edit]
user@host# set protocols ospf area 1 interface at-0/1/1.100
[edit]
user@host# show protocols ospf
ospf {
        area 0.0.0.0 {
                interface ge-0/0/0.0;
        }
        area 0.0.0.1 {
                interface at-0/1/1.100;
        }
```

8.3.2 配置 a Stub Area

```
[edit protocols ospf area area-id ]
stub <default-metric metric> <(no-summaries | summaries)>;
```

8.3.3 配置 a Not-So-Stubby Area

```
[edit protocols ospf area area-id ]
nssa {
```

area-range *network/mask-length* <restrict>;

default-lsa {

default-metric *metric*;

metric-type *type*;

type-7;

}

(no-summaries | summaries);

}

8.4.4 配置 OSPF Virtual Link

当网络直接超过3 area，使用virtual Link 连接防止环路。

[edit protocols ospf area 0.0.0.0]

virtual-link neighbor-id *router-id* transit-area *area-id*;

8.3.5 配置 OSPF Router Interfaces

- **Configuring an Interface on a Broadcast or Point-to-Point Network**

[edit protocols ospf area *area-id*]

interface *interface-name*;

- **Configuring an Interface on a Point-to-Multipoint Network**

[edit protocols ospf area 0.0.0.0]

interface *interface-name* {

neighbor *address*;

}

- **Configuring an Interface on a Nonbroadcast, Multiaccess Network**

[edit protocols ospf area 0]

interface *interface-name* {

interface-type nbma;

neighbor *address* <eligible>;

poll-interval *seconds*;

}

8.3.6 配置验证

可以使**simple**和**MD5**

[edit protocols ospf area *area-id*]

authentication-type *authentication*;

[edit protocols ospf area *area-id* interface *interface-name*]
authentication {
md5 *key-id* {
key [*key-values*];
}
simple-password *key-id*;
}

8.4 JUNOS Troubleshooting

8.4.1 查看参与 OSPF 运行的接口 show ospf interface

```
user@Chardonnay> show ospf interface
Interface       State    Area          DR ID          BDR ID         Nbrs
at-0/1/0.100    PtToPt   0.0.0.0       0.0.0.0        0.0.0.0           1
so-0/0/1.0      PtToPt   0.0.0.0       0.0.0.0        0.0.0.0           1
so-0/0/0.0      PtToPt   0.0.0.10      0.0.0.0        0.0.0.0           1
so-0/0/2.0      PtToPt   0.0.0.10      0.0.0.0        0.0.0.0           1
so-0/0/3.0      PtToPt   0.0.0.10      0.0.0.0        0.0.0.0           1
```

Area — This field displays the current area ID assigned to the interface.

DR ID — The router ID of the current designated router is displayed in this column. Point-to-point interfaces use a value of 0.0.0.0.

BDR ID — The router ID of the current backup designated router is displayed in this column. Point-to-point interfaces use a value of 0.0.0.0.

Nbrs — The value in this column represents the total number of OSPF neighbors discovered across this interface.

8.4.2 查看 OSPF 邻居

```
user@Chardonnay> show ospf neighbor
  Address          Interface        State      ID              Pri   Dead
10.0.1.46        at-0/1/0.100     Full       10.0.1.103      128    36
10.0.1.34        so-0/0/1.0       Full       10.0.1.102      128    35
10.0.1.9         so-0/0/0.0       Full       10.0.1.21       128    38
```

```
10.0.1.5         so-0/0/2.0        Full       10.0.1.22          128    32
10.0.1.1         so-0/0/3.0        Full       10.0.1.23          128    39
```

Address The physical interface IP address of the neighbor is displayed in this column.

Interface This column shows the OSPF interface that the neighbor is reachable across.

State The current OSPF adjacency state is displayed here. The possible state values are discussed in the "Forming Adjacencies" section earlier in this chapter.

ID This field shows the router ID of the neighbor. This is used with the Pri field to elect a DR or BDR on a broadcast segment.

Pri The router priority is displayed in this field. This value is used with the ID field to elect a DR or BDR on a broadcast or NBMA segment.

Dead The time remaining until the OSPF neighbor is declared unreachable appears in this column. Each received hello packet resets this timer to the router dead interval value.

8.4.3 查看 OSPF LSA 数据库

show ospf database

```
user@Shiraz> show ospf database

    OSPF link state database, area 0.0.0.10
 Type       ID          Adv Rtr            Seq          Age  Opt  Cksum  Len
Router   *10.0.1.21    10.0.1.21          0x80000004   2965  0x2  0x3407   60
Router    10.0.1.22    10.0.1.22          0x80000004   2971  0x2  0xb58a   60
Router    10.0.1.23    10.0.1.23          0x80000008   2800  0x2  0x2f12   60
Router    10.0.1.101   10.0.1.101         0x8000000c   1328  0x2  0x6d4   108
Summary   10.0.1.0     10.0.1.101         0x80000005    728  0x2  0x3525   28
ASBRSum   10.0.1.105   10.0.1.101        0x80000006    128  0x2  0xf976   28
    OSPF external link state database
 Type       ID             Adv Rtr            Seq          Age  Opt  Cksum  Len
Extern    192.168.1.0    10.0.1.105         0x80000034    306  0x2  0xe5da   36
Extern    192.168.2.0    10.0.1.105         0x80000034      5  0x2  0xdae4   36
Extern    192.168.3.0    10.0.1.105         0x80000033   1206  0x2  0xd1ed   36
Extern    192.168.4.0    10.0.1.105         0x80000033    907  0x2  0xc6f7   36
```

- Router-Type 1 router LSA
- Network-Type 2 network LSA
- Summary-Type 3 network summary LSA
- ASBRSum-Type 4 ASBR summary LSA
- Extern-Type 5 AS external LSA
- NSSA-Type 7 NSSA external LSA

ID This field shows the Link-State ID field from the LSA. This value is used to provide uniqueness for each LSA. Entries marked with an asterisk (*) are LSAs generated by the local router.

Adv Rtr The router ID of the originating router for each LSA is displayed in this field.

Seq The sequence number assists the router to determine the most recent version of the LSA.

Age This field displays the current age of the LSA. All LSAs begin with a lifetime of 0 and increment to a defined MaxAge of 3600 seconds. Each LSA must be refreshed before the MaxAge value is reached.

Opt The Options field from the OSPF header is displayed in this column. The possible bit values are discussed in the "Hello Packet" section earlier in this chapter.

Cksum The calculated checksum value of the LSA is stored in this field. Each router calculates a new checksum when the LSA is received and verifies the value against the received value to ensure packet integrity.

Len This field displays the total length of the LSA.

8.4.3 清空 OSPF 数据库

clear ospf database

user@Shiraz> clear ospf database purge

手工清空 OSPF floods 的数据库。

8.4.5 查看 OSPF Log

user@Shiraz> show ospf log

```
   Last instance of each event type
When            Type            Elapsed
00:17:29        SPF             0.000073
00:17:29        Stub            0.000067
00:17:29        Interarea       0.000025
00:17:29        External        0.000003
00:17:29        NSSA            0.000003
00:17:29        Cleanup         0.000083

   Maximum length of each event type
When            Type            Elapsed
01:17:57        SPF             0.000116
00:22:41        Stub            0.000365
20:00:18        Interarea       0.000132
01:19:43        External        0.000042
19:17:29        NSSA            0.000014
19:17:29        Cleanup         0.000715

   Last 100 events
When            Type            Elapsed
01:19:48          Total         0.000182
01:19:43        SPF             0.000090
01:19:43        Stub            0.000086
01:19:43        Interarea       0.000030
01:19:43        External        0.000042
01:19:43        NSSA            0.000004
...[output truncated]
```

8.4.6 查看 OSPF statistics

```
user@Shiraz> show ospf statistics

Packet type             Total                 Last 5 seconds
                   Sent      Received        Sent      Received
   Hello             24            45           0             0
     DbD             24            16           0             0
   LSReq              6             7           0             0
LSUpdate            375          2260           0             0
   LSAck           2236           368           0             0
...
```

8.4.7 查看运行 OSPF 算法

show ospf route

```
user@Chardonnay> show ospf route
Prefix                Path   Route        NH    Metric  NextHop
Nexthop
                      Type   Type         Type          Interface
addr/label
10.0.1.21/32          Intra  Router       IP     1       so-0/0/0.0
10.0.1.102/32         Intra  Router       IP     1       so-0/0/1.0
10.0.1.103/32         Intra  Router       IP     1       at-0/1/0.100
10.0.1.104/32         Intra  Router       IP     2       at-0/1/0.100
                                                        so-0/0/1.0
10.0.1.105/32         Intra  AS BR        IP     2       so-0/0/1.0
10.0.1.106/32         Intra  Router       IP     3       at-0/1/0.100
                                                        so-0/0/1.0
```

8.4.8 查看 ospf 路由

```
user@Chardonnay> show route protocol ospf

inet.0: 34 destinations, 40 routes (34 active, 0 holddown, 0 hidden)
+ = Active Route, - = Last Active, * = Both
10.0.1.0/30         [OSPF/10] 03:02:40, metric 1
                     > via so-0/0/3.0
10.0.1.4/30         [OSPF/10] 03:02:40, metric 1
                     > via so-0/0/2.0
10.0.1.8/30         [OSPF/10] 03:02:40, metric 1
                     > via so-0/0/0.0
10.0.1.21/32       *[OSPF/10] 02:25:42, metric 1
                     > via so-0/0/0.0
10.0.1.32/30        [OSPF/10] 03:02:40, metric 1
                     > via so-0/0/1.0
192.168.1.0/24     *[OSPF/150] 03:02:40, metric 0, tag 0
                     > via so-0/0/1.0
192.168.2.0/24     *[OSPF/150] 03:02:40, metric 0, tag 0
                     > via so-0/0/1.0
192.168.3.0/24     *[OSPF/150] 03:02:40, metric 0, tag 0
                     > via so-0/0/1.0
192.168.4.0/24     *[OSPF/150] 03:02:40, metric 0, tag 0
                     > via so-0/0/1.0
224.0.0.5/32       *[OSPF/10] 5d 17:20:57, metric 1
```

8.5 配置 OSPF 多区域

如图：互连地址 192.168.101.x/30 网段。 网络 lo0：192.168.102.x/32。

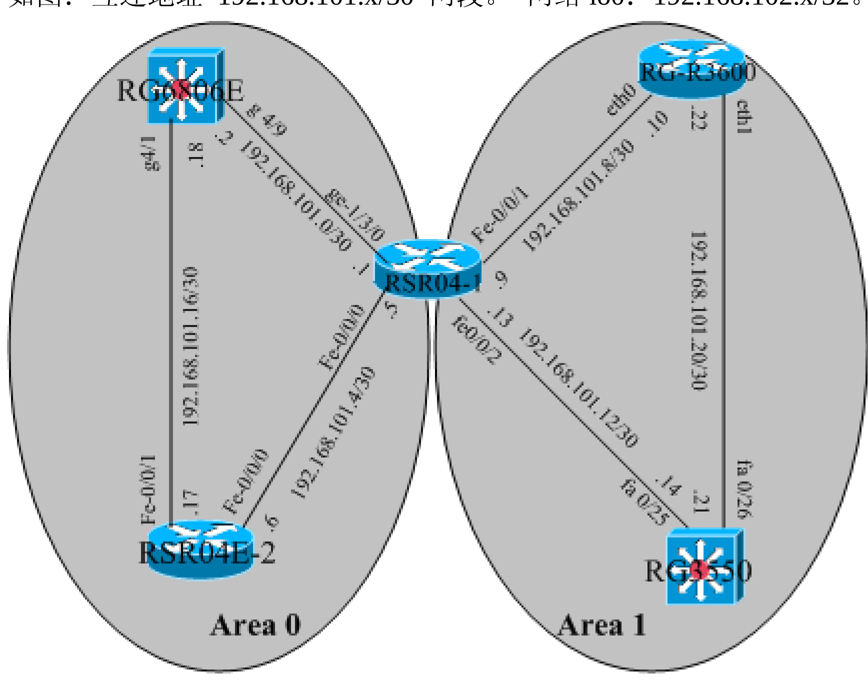


图 8 - 6

配置脚本：

RSR04E-1

```
interfaces {
        fe-0/0/0 {
                description to-RSR04E-2-fe-0/0/0;
                unit 0 {
                        family inet {
                                address 192.168.101.5/30;
                        }
                }
        }
        fe-0/0/1 {
                description to-RG-R3600-eth0;
                unit 0 {
                        family inet {
                                address 192.168.101.9/30;
                        }
```

```
            }
        }
        fe-0/0/2 {
            description "to-RG-S3550-fa 0/25";
            unit 0 {
                family inet {
                    address 192.168.101.13/30;
                }
            }
        }
        ge-1/3/0 {
            description to-RG-S6806E-g4/9;
            unit 0 {
                family inet {
                    address 192.168.101.1/30;
                }
            }
        }
        lo0 {
            description RID;
            unit 0 {
                family inet {
                    address 192.168.102.1/32;
                }
            }
        }
}
routing-options {
        router-id 192.168.102.1;
}
protocols {
        ospf {
            area 0.0.0.0 {
                interface lo0.0 {
                    passive;
                }
                interface ge-1/3/0.0;
                interface fe-0/0/0.0;
            }
            area 0.0.0.1 {
                interface fe-0/0/1.0;
                interface fe-0/0/2.0;
            }
        }
```

```
}
```

RSR04E-2：

```
interfaces {
    fe-0/0/0 {
        description to-RSR04E-1-fe-0/0/0;
        speed 100m;
        link-mode full-duplex;
        unit 0 {
            family inet {
                address 192.168.101.6/30;
            }
        }
    }
    fe-0/0/1 {
        description to-RG6806E-g4/1;
        unit 0 {
            family inet {
                address 192.168.101.17/30;
            }
        }
    }
    lo0 {
        description RID;
        unit 0 {
            family inet {
                address 192.168.102.2/32;
            }
        }
    }
}
routing-options {
    router-id 192.168.102.2;
}
protocols {
    ospf {
        area 0.0.0.0 {
            interface lo0.0 {
                passive;
            }
```

```
                interface fe-0/0/0.0;
                interface fe-0/0/1.0;
            }
        }
}
```

RG-6806E：

```
interface GigabitEthernet 4/1
  no switchport
  description "to-RSR04E-2-fe-0/0/1"
  ip address 192.168.101.18 255.255.255.252
!
interface GigabitEthernet 4/9
  medium-type fiber
  no switchport
  speed    1000
  description "to-RSR04E-ge-1/3/0"
  ip address 192.168.101.2 255.255.255.252
!
interface Loopback 0
   description "RID"
  ip address 192.168.102.5 255.255.255.255
!
router ospf
area 0.0.0.0
network 192.168.101.0 255.255.255.252 area 0.0.0.0
network 192.168.101.16 255.255.255.252 area 0.0.0.0
network 192.168.102.5 255.255.255.255 area 0.0.0.0
!
End
```

RG3660：

```
!
interface FastEthernet 0/0
  ip address 192.168.101.10 255.255.255.252
  duplex auto
  speed auto
  description to-RSR04E-1-fe-0/0/1
!
interface FastEthernet 0/1
  ip address 192.168.101.22 255.255.255.252
  duplex auto
  speed auto
```

```
 no keepalive
 description to-RG-S3550-fa-0/25
!
interface Loopback 0
 ip address 192.168.102.3 255.255.255.255
 description RID
!
router ospf
 passive-interface Loopback 0
 network 192.168.101.8 0.0.0.3 area 0.0.0.1
 network 192.168.101.20 0.0.0.3 area 0.0.0.1
 network 192.168.102.3 0.0.0.0 area 0.0.0.1
!
```

RG-S3550：

```
!
interface FastEthernet 0/25
 no switchport
 description "to-RSR04E-1-fe-0/0/2"
 ip address 192.168.101.14 255.255.255.252
!
interface FastEthernet 0/26
 no switchport
 description "to-RG-R3600-fa0/1"
 ip address 192.168.101.21 255.255.255.252
!
interface Loopback 0
 description "RID"
 ip address 192.168.102.4 255.255.255.255
!
router ospf
area 0.0.0.1
network 192.168.101.12 255.255.255.252 area 0.0.0.1
network 192.168.101.20 255.255.255.252 area 0.0.0.1
network 192.168.102.4 255.255.255.255 area 0.0.0.1
!
End
```

8.5.1 查看 RSR ospf 学习的路由

admin@RSR04E-1> **show route protocol ospf**

inet.0: 18 destinations, 18 routes (18 active, 0 holddown, 0 hidden)

```
+ = Active Route, - = Last Active, * = Both

192.168.101.16/30    *[OSPF/10] 00:35:03, metric 2
                       > to 192.168.101.6 via fe-0/0/0.0
                         to 192.168.101.2 via ge-1/3/0.0
192.168.101.20/30    *[OSPF/10] 00:27:05, metric 2
                       > to 192.168.101.10 via fe-0/0/1.0
                         to 192.168.101.14 via fe-0/0/2.0
192.168.102.2/32     *[OSPF/10] 00:35:03, metric 1
                       > to 192.168.101.6 via fe-0/0/0.0
192.168.102.3/32     *[OSPF/10] 00:27:05, metric 2
                       > to 192.168.101.10 via fe-0/0/1.0
192.168.102.4/32     *[OSPF/10] 00:46:39, metric 2
                       > to 192.168.101.14 via fe-0/0/2.0
192.168.102.5/32     *[OSPF/10] 00:46:50, metric 2
                       > to 192.168.101.2 via ge-1/3/0.0
224.0.0.5/32         *[OSPF/10] 01:13:33, metric 1
                         MultiRecv
```

8.5.2 查看 OSPF 邻接关系

admin@RSR04E-1> **show ospf neighbor**

Address	Interface	State	ID	Pri	Dead
192.168.101.6	fe-0/0/0.0	Full	192.168.102.2	128	35
192.168.101.2	ge-1/3/0.0	Full	192.168.102.5	1	39
192.168.101.10	fe-0/0/1.0	Full	192.168.102.3	1	30
192.168.101.14	fe-0/0/2.0	Full	192.168.102.4	1	30

8.5.3 查看参与 ospf 接口

admin@RSR04E-1> **show ospf interface**

Interface	State	Area	DR ID	BDR ID	Nbrs
fe-0/0/0.0	DR	0.0.0.0	192.168.102.1	192.168.102.2	1
ge-1/3/0.0	DR	0.0.0.0	192.168.102.1	192.168.102.5	1
lo0.0	DRother	0.0.0.0	0.0.0.0	0.0.0.0	0
fe-0/0/1.0	BDR	0.0.0.1	192.168.102.3	192.168.102.1	1
fe-0/0/2.0	BDR	0.0.0.1	192.168.102.4	192.168.102.1	1

8.5.4 查看路由表摘要

admin@RSR04E-1> **show route protocol ospf | match /32**
192.168.102.2/32 *[OSPF/10] 00:55:06, metric 1
192.168.102.3/32 *[OSPF/10] 00:55:06, metric 2
192.168.102.4/32 *[OSPF/10] 00:55:06, metric 2
192.168.102.5/32 *[OSPF/10] 00:55:06, metric 2
224.0.0.5/32 *[OSPF/10] 02:31:34, metric 1

8.5.5 查看转发表

admin@RSR04E-1> **show route forwarding-table destination 192.168.101.16/30**
Routing table: inet
Internet:
Destination Type RtRef Next hop Type Index NhRef Netif
192.168.101.16/30 user 0 192.168.101.6 ucst 341 3 fe-0/0/0.0

注意没有形成负载均衡，junos 默认是没有的，可以配置，查看前面关于负载均衡章节（5.7）。

8.5.6 查看接口 cost

请注意由于 ospf interface 参考值 100M 和 1000M cost 都是等于 1，所以在 RSR04E-1 上网段 192.168.0.16/30 可以负载均衡。通过在全网通过 **set protocols ospf reference-bandwidth < value>** 更准确的描述 ospf cost 参考值。

admin@RSR04E-1> **show ospf interface fe-0/0/0.0 extensive**
Interface State Area DR ID BDR ID Nbrs
fe-0/0/0.0 DR 0.0.0.0 192.168.102.1 192.168.102.2 1
Type: LAN, Address: 192.168.101.5, Mask: 255.255.255.252, MTU: 1500, **Cost: 1**
DR addr: 192.168.101.5, BDR addr: 192.168.101.6, Adj count: 1, Priority: 128
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None

admin@RSR04E-1> **show ospf interface ge-1/3/0.0 extensive**
Interface State Area DR ID BDR ID Nbrs
ge-1/3/0.0 DR 0.0.0.0 192.168.102.1 192.168.102.5 1
Type: LAN, Address: 192.168.101.1, Mask: 255.255.255.252, MTU: 1500, **Cost: 1**
DR addr: 192.168.101.1, BDR addr: 192.168.101.2, Adj count: 1, Priority: 128
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: None

8.5.7 OSPF Authentication

- JUNOS 支持 MD5 和 Simple 密码验证。

Example:

拓扑如图：8.5-1 Area 0 用 MD5 验证。Area 1 使用明文验证。

```
dmin@RSR04E-1# show
ospf {
    area 0.0.0.0 {
        authentication-type md5;
        interface lo0.0 {
            passive;
        }
        interface ge-1/3/0.0 {
            authentication {
                md5 10 key "$9$nj7z9tOhSeX7V1R7VwYZG69A"; ## SECRET-DATA
            }
        }
        interface fe-0/0/0.0 {
            authentication-type simple;
            authentication {
                md5 10 key "$9$tXIZ01hevLVwgSrwgoJHkp0B"; ## SECRET-DATA
            }
        }
    }
    area 0.0.0.1 {
        interface fe-0/0/1.0 {
            authentication {
                simple-password "$9$WlEXNb4aU.PQaZ6A"; ## SECRET-DATA
            }
        }
        interface fe-0/0/2.0 {
            authentication {
                simple-password "$9$ZSDHmz39O1h36lM"; ## SECRET-DATA
            }
        }
    }
}
```

```
RG-6806E#show run
Building configuration...
Current configuration : 1005 bytes

!
```

version 1.0
install 4 12sfp/gt
ip routing algorithm CRC32_UPPER
!
hostname RG-6806E
enable secret level 1 5 $2lowNq&3h`@IOrJ4imLMp]KQknAxB^"
enable secret level 15 5 $2,1u_;C3&-8U0<D4'.tj9=GQ+/7R:>H
!
interface GigabitEthernet 4/1
  no switchport
  description "to-RSR04E-2-fe-0/0/1"
  ip address 192.168.101.18 255.255.255.252
ip ospf authentication message-digest
ip ospf message-digest-key 10 md5 juniper
!
interface GigabitEthernet 4/9
  medium-type fiber
  no switchport
  speed   1000
  description "to-RSR04E-ge-1/3/0"
  ip address 192.168.101.2 255.255.255.252
ip ospf authentication message-digest
ip ospf message-digest-key 10 md5 juniper
!
interface Loopback 0
  description "RID"
  ip address 192.168.102.5 255.255.255.255
!
router ospf
area 0.0.0.0
area 0.0.0.0 authentication message-digest
network 192.168.101.0 255.255.255.252 area 0.0.0.0
network 192.168.101.16 255.255.255.252 area 0.0.0.0
network 192.168.102.5 255.255.255.255 area 0.0.0.0
!
End
RG3660#show run

Building configuration...
Current configuration : 988 bytes

!
version 8.2(building 11)
hostname RG3660

```
enable password 7 1316064b1f
!
interface FastEthernet 0/0
 ip ospf authentication-key junos
 ip address 192.168.101.10 255.255.255.252
 duplex auto
 speed auto
 description to-RSR04E-1-fe-0/0/1
!
interface FastEthernet 0/1
 ip ospf authentication-key junos
 ip address 192.168.101.22 255.255.255.252
 duplex auto
 speed auto
 no keepalive
 description to-RG-S3550-fa-0/25
!
interface FastEthernet 6/0
 duplex auto
 speed auto
!
interface FastEthernet 6/1
 duplex auto
 speed auto
!
interface Loopback 0
 ip address 192.168.102.3 255.255.255.255
 description RID
!
interface Null 0
!
!
router ospf
 passive-interface Loopback 0
 area 0.0.0.1 authentication
 network 192.168.101.8 0.0.0.3 area 0.0.0.1
 network 192.168.101.20 0.0.0.3 area 0.0.0.1
 network 192.168.102.3 0.0.0.0 area 0.0.0.1
!
!
!
!
line con 0
line aux 0
```

line vty 0 4
 no login
!
!
End

● 查看接口状态：

admin@RSR04E-1> show ospf interface

Interface	State	Area	DR ID	BDR ID	Nbrs
ge-1/3/0.0	DR	0.0.0.0	192.168.102.1	192.168.102.5	1

Type: LAN, Address: 192.168.101.1, Mask: 255.255.255.252, MTU: 1500, Cost: 1
DR addr: 192.168.101.1, BDR addr: 192.168.101.2, Adj count: 1, Priority: 128
Hello: 10, Dead: 40, ReXmit: 5, Not Stub

● **Auth type: MD5, Active key ID: 10, Start time: 1970 Jan 1 00:00:00 UTC**

admin@RSR04E-1# run show ospf interface fe-0/0/1.0 extensive

Interface	State	Area	DR ID	BDR ID	Nbrs
fe-0/0/1.0	DR	0.0.0.1	192.168.102.1	192.168.102.3	1

Type: LAN, Address: 192.168.101.9, Mask: 255.255.255.252, MTU: 1500, Cost: 1
DR addr: 192.168.101.9, BDR addr: 192.168.101.10, Adj count: 1, Priority: 128
Hello: 10, Dead: 40, ReXmit: 5, Not Stub
Auth type: Password

● 通过 **monitor** 查看接口

admin@RSR04E-1> monitor traffic interface ge-1/3/0.0 extensive
Listening on ge-1/3/0.0, capture size 96 bytes

07:08:49.412960 In (tos 0xc0, ttl 1, id 3390, offset 0, flags [none], proto: OSPF (89), length: 84) 192.168.101.2 > 224.0.0.5: OSPFv2, Hello (1), length: 48
    Router-ID: 192.168.102.5, Backbone Area, **Authentication Type: MD5 (2)**
    Key-ID: 10, Auth-Length: 16, Crypto Sequence Number: 0x0000017a
    Options: [External]
      Hello Timer: 10s, Dead Timer 40s, Mask: 255.255.255.252, Priority: 1
      Designated Router 192.168.101.1, Backup Designated Router 192.168.101.2
      Neighbor List:
        192.168.102.1

07:08:56.740876 Out 0:12:1e:1:30:db 1:0:5e:0:0:5 ip 98: (tos 0xc0, ttl 1, id 52302, offset 0, flags [none], proto: OSPF (89), length: 84) 192.168.101.1 > 224.0.0.5: OSPFv2, Hello (1), length: 48
    Router-ID: 192.168.102.1, Backbone Area, **Authentication Type: MD5 (2)**
    Key-ID: 10, Auth-Length: 16, Crypto Sequence Number: 0x000054b6
    Options: [External]
      Hello Timer: 10s, Dead Timer 40s, Mask: 255.255.255.252, Priority: 128
      Designated Router 192.168.101.1, Backup Designated Router 192.168.101.2
      Neighbor List: [|ospf]

admin@RSR04E-1> monitor traffic interface fe-0/0/1.0 extensive

Listening on fe-0/0/1.0, capture size 96 bytes

07:12:19.423729 In (tos 0xc0, ttl 1, id 16641, offset 0, flags [none], proto: OSPF (89), length: 68) 192.168.101.10 > 224.0.0.5: OSPFv2, Hello (1), length: 48

Router-ID: 192.168.102.3, Area 0.0.0.1, **Authentication Type:** unknown (1)junos^@^@^@"

Options: [External]

Hello Timer: 10s, Dead Timer 40s, Mask: 255.255.255.252, Priority: 1

Designated Router 192.168.101.9, Backup Designated Router 192.168.101.10

Neighbor List:

192.168.102.1

07:12:20.892801 Out 0:12:1e:1:30:1 1:0:5e:0:0:5 ip 82: (tos 0xc0, ttl 1, id 52543, offset 0, flags [none], proto: OSPF (89), length: 68) 192.168.101.9 > 224.0.0.5: OSPFv2, Hello (1), length: 48

Router-ID: 192.168.102.1, Area 0.0.0.1, **Authentication Type:** unknown (1)junos^@^@^@"

Options: [External]

Hello Timer: 10s, Dead Timer 40s, Mask: 255.255.255.252, Priority: 128

Designated Router 192.168.101.9, Backup Designated Router 192.168.101.10

Neighbor List:

192.168.102.3

8.6 配置 Stub Areas

Stub 区域: LSA types 5 and 7 不存在 Stub Areas，没有 LSA types 5 自然 ASBR 自然不会去通告 LSA type 4。

配置拓扑如图：

Area 10 为 stub area

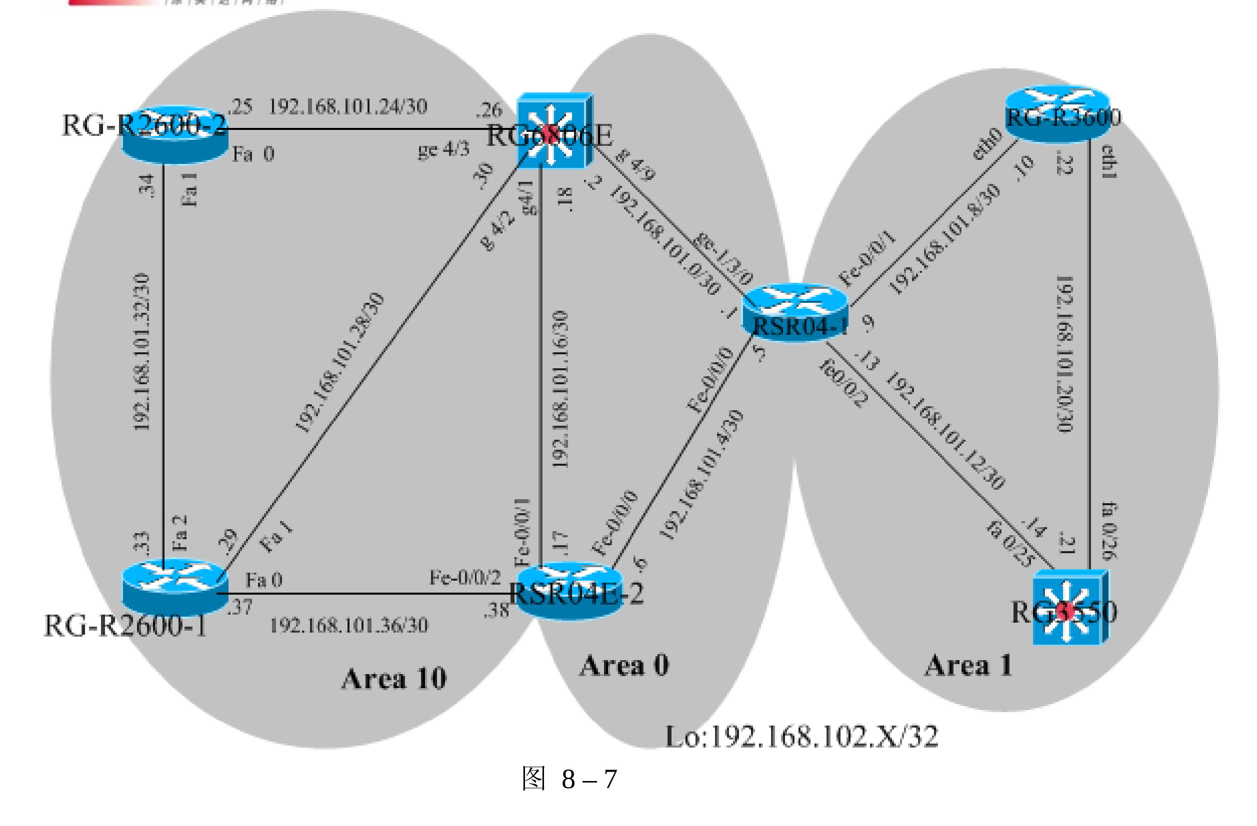


图 8 – 7

通过在 RSR04E-1 在发布网段 200.0.0.0/24 到 ospf 增加 ospf 类型 LSA 5 type 2

- 添加一条静态路由

```
admin@RSR04E-1# set routing-options static route 200.0.0.0/24 reject
[edit]
```

匹配策略

```
policy-options {
    policy-statement static {
        term static {
            from {
                protocol static;
                route-filter 200.0.0.0/24 exact;
            }
            then {
                metric 111;
                external {
                    type 2;
                }
                accept;
            }
        }
        term deny-any {
            then reject;
```

```
        }
    }
}
```

- 把 route 发布到 ospf ；

```
protocols {
        export static;
}
RG-R2600-2#sho ip route
Codes: C - connected, S - static,    R - RIP
       O - OSPF, IA - OSPF inter area
       E1 - OSPF external type 1, E2 - OSPF external type 2

Gateway of last resort is not set

     192.168.101.0/30 is subnetted, 10 subnets
O IA     192.168.101.12 [110/3] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.101.8 [110/3] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.101.4 [110/3] via 192.168.101.33, 00:20:54, FastEthernet1
                       [110/3] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.101.0 [110/2] via 192.168.101.26, 00:20:54, FastEthernet0
O        192.168.101.28 [110/2] via 192.168.101.26, 00:45:12, FastEthernet0
                        [110/2] via 192.168.101.33, 00:45:12, FastEthernet1
C        192.168.101.24 is directly connected, FastEthernet0
O IA     192.168.101.20 [110/4] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.101.16 [110/2] via 192.168.101.26, 00:20:54, FastEthernet0
O        192.168.101.36 [110/2] via 192.168.101.33, 00:45:12, FastEthernet1
C        192.168.101.32 is directly connected, FastEthernet1
     192.168.102.0/32 is subnetted, 7 subnets
O        192.168.102.6 [110/2] via 192.168.101.33, 00:45:12, FastEthernet1
C        192.168.102.7 is directly connected, Loopback0
O IA     192.168.102.4 [110/4] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.102.5 [110/2] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.102.2 [110/2] via 192.168.101.33, 00:20:54, FastEthernet1
                       [110/2] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.102.3 [110/4] via 192.168.101.26, 00:20:54, FastEthernet0
O IA     192.168.102.1 [110/2] via 192.168.101.26, 00:20:54, FastEthernet0
O E2 200.0.0.0/24 [110/111] via 192.168.101.26, 00:13:48, FastEthernet0
```

- **通过 配置 area 10 为 stub 去处 LSA 5**

admin@RSR04E-2# **set protocols ospf area 10 stub**

RG-R2600-2# show ip route
Codes: C - connected, S - static,   R - RIP
            O - OSPF, IA - OSPF inter area
            E1 - OSPF external type 1, E2 - OSPF external type 2

Gateway of last resort is 192.168.101.26 to network 0.0.0.0

        192.168.101.0/30 is subnetted, 10 subnets
O IA   192.168.101.12 [110/3] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.101.8 [110/3] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.101.4 [110/3] via 192.168.101.33, 00:00:55, FastEthernet1
                        [110/3] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.101.0 [110/2] via 192.168.101.26, 00:00:55, FastEthernet0
O   192.168.101.28 [110/2] via 192.168.101.33, 00:00:55, FastEthernet1
                        [110/2] via 192.168.101.26, 00:00:55, FastEthernet0
C   192.168.101.24 is directly connected, FastEthernet0
O IA   192.168.101.20 [110/4] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.101.16 [110/2] via 192.168.101.26, 00:00:55, FastEthernet0
O   192.168.101.36 [110/2] via 192.168.101.33, 00:00:55, FastEthernet1
C   192.168.101.32 is directly connected, FastEthernet1
        192.168.102.0/32 is subnetted, 7 subnets
O   192.168.102.6 [110/2] via 192.168.101.33, 00:00:55, FastEthernet1
C   192.168.102.7 is directly connected, Loopback0
O IA   192.168.102.4 [110/4] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.102.5 [110/2] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.102.2 [110/2] via 192.168.101.33, 00:00:55, FastEthernet1
                        [110/2] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.102.3 [110/4] via 192.168.101.26, 00:00:55, FastEthernet0
O IA   192.168.102.1 [110/2] via 192.168.101.26, 00:00:55, FastEthernet0

没有了 **O E2 200.0.0.0/24 [110/111] via 192.168.101.26, 00:13:48, FastEthernet0**

- 通过添加 **no-summaries** 在 **area 10** 内没有 **LSA types 3 and 4** 。只有一个 **default route 0.0.0.0**

admin@RSR04E-2# set protocols ospf area 10 stub no-summaries default-metric 1

RG-R2600-2#sho ip route
Codes: C - connected, S - static,   R - RIP
            O - OSPF, IA - OSPF inter area
            E1 - OSPF external type 1, E2 - OSPF external type 2

Gateway of last resort is 192.168.101.26 to network 0.0.0.0

        192.168.101.0/30 is subnetted, 4 subnets

```
O           192.168.101.28 [110/2] via 192.168.101.33, 00:06:03, FastEthernet1
                           [110/2] via 192.168.101.26, 00:06:03, FastEthernet0
C           192.168.101.24 is directly connected, FastEthernet0
O           192.168.101.36 [110/2] via 192.168.101.33, 00:06:03, FastEthernet1
C           192.168.101.32 is directly connected, FastEthernet1
      192.168.102.0/32 is subnetted, 2 subnets
O           192.168.102.6 [110/2] via 192.168.101.33, 00:06:03, FastEthernet1
C           192.168.102.7 is directly connected, Loopback0
O*IA 0.0.0.0/0 [110/2] via 192.168.101.26, 00:04:50, FastEthernet0
```

- 查看 **ospf database**

RG-R2600-2#**sho ip ospf database**

OSPF Router with ID (192.168.101.34) (Process ID 10)

OSPF Router with ID (192.168.102.7) (Process ID 100)

Router Link States (Area 10)

Link ID	ADV Router	Age	Seq#	Checksum	Link count
192.168.102.2	192.168.102.2	328	0x8000000B	0xC4F8	1
192.168.102.5	192.168.102.5	813	0x80000015	0xD507	2
192.168.102.6	192.168.102.6	404	0x80000013	0xD718	4
192.168.102.7	192.168.102.7	810	0x8000000D	0x38B9	3

Net Link States (Area 10)

Link ID	ADV Router	Age	Seq#	Checksum
192.168.101.25	192.168.102.7	950	0x80000002	0x945F
192.168.101.29	192.168.102.6	1103	0x80000002	0x6889
192.168.101.34	192.168.102.7	950	0x80000002	0x48A1
192.168.101.37	192.168.102.6	587	0x80000001	0xEFFD

Summary Net Link States (Area 10)

Link ID	**ADV Router**	**Age**	**Seq#**	**Checksum**
0.0.0.0	**192.168.102.2**	**328**	**0x80000001**	**0xAEDD**
0.0.0.0	**192.168.102.5**	**1546**	**0x80000001**	**0x9CEC**

。。。

8.7 配置 Not-So-Stubby Areas (NSSA)

No type 5 or type 3 LSAs 减小 NSSA,在边界路由器 ABR 上产生把 LSA type 7 转换成 LSA 5 发布到其它 area。

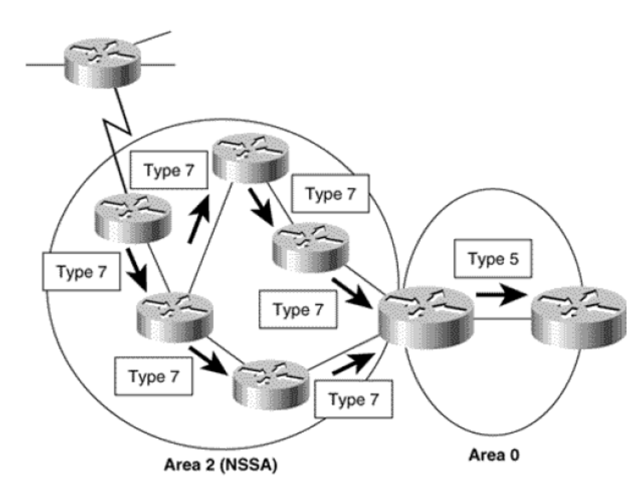


图 8 – 8

NSSA 配置命令

```
edit protocols ospf area 0.0.0.10]
admin@RSR04E-2#show
nssa {
  default-lsa {
    default-metric <value>;
    metric-type <value>;
    type-7;
  }
  no-summaries;
}
interface <interface-name>;

}
```

8.8 配置 Virtual Links

Virtual Links — 为防止出现非连续的 backbone area （area 0），形成环路。
在下拓扑中由于 RSR04E-2 与 RSR04E-1 的链路断开，造成 RSR04E-2 的 lo0:
在 area 0 中。出现两不连续的网段。所以配置拓扑。如图 8.8-1 和图 8.8-2 拓扑

Syntax:

```
protocols {
    ospf {
        area 0.0.0.0 {
            authentication-type md5;
            virtual-link neighbor-id 192.168.102.5 transit-area 0.0.0.10 {
                authentication {
                    md5 10 key "$9$FFE56CuRhr8X-O1X-VwaJ369"; ## SECRET-DATA
                }
            }
```

图 8.8-1

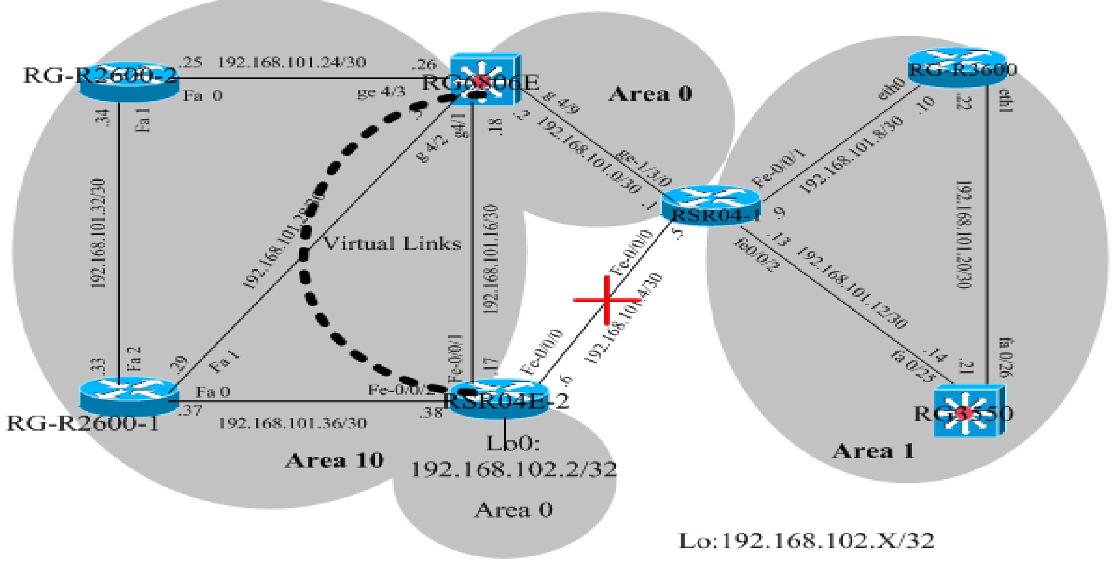


图 8.8-2

配置如下：

RSR04E-2

```
protocols {
    ospf {
        area 0.0.0.0 {
```

```
                authentication-type md5;
                virtual-link neighbor-id 192.168.102.5 transit-area 0.0.0.10 {
                    authentication {
                        md5 10 key "$9$FFE56CuRhr8X-O1X-VwaJ369"; ## SECRET-DATA
                    }
                }
                interface lo0.0 {
                    passive;
                }
                interface fe-0/0/0.0 {
                    authentication {
                        md5 10 key "$9$oGZDk5Qnp0I.P0IEcvMaZU"; ## SECRET-DATA
                    }
                }
            }
            area 0.0.0.10 {
                interface fe-0/0/2.0;
                interface fe-0/0/1.0;
            }
        }
}
```

RG-R6806E

```
admin@RSR04E-1# run telnet 192.168.102.5
Trying 192.168.102.5...
Connected to 192.168.102.5.
Escape character is '^]'.


User Access Verification

Password:
RG-6806E>en
Password:

RG-6806E#sho run
Building configuration...
Current configuration : 1501 bytes

!
version 1.0
install 4 12sfp/gt
ip routing algorithm CRC32_UPPER
!
hostname RG-6806E
```

```
enable secret level 1 5 $2-aeh`@31'dfimL4t{bcknAQ7zyglow
enable secret level 15 5 $2.Y*T7+3,tZ[V/,4+S(\W&-Q1X)sv'~
!
interface GigabitEthernet 4/1
 no switchport
 description "to-RSR04E-2-fe-0/0/1"
 ip address 192.168.101.18 255.255.255.252
!
interface GigabitEthernet 4/2
 no switchport
 description "to-RG-R2600-1-fa/1"
 ip address 192.168.101.30 255.255.255.252
!
interface GigabitEthernet 4/3
 no switchport
 description "to-GR-R2600-2-fa/0"
 ip address 192.168.101.26 255.255.255.252
!
interface GigabitEthernet 4/9
 medium-type fiber
 no switchport
 speed   1000
 description "to-RSR04E-ge-1/3/0"
 ip address 192.168.101.2 255.255.255.252
ip ospf authentication message-digest
ip ospf message-digest-key 10 md5 juniper
!
interface Loopback 0
 description "RID"
 ip address 192.168.102.5 255.255.255.255
!
router ospf
area 0.0.0.0
area 0.0.0.0 authentication message-digest
area 0.0.0.10
network 192.168.101.0 255.255.255.252 area 0.0.0.0
network 192.168.102.5 255.255.255.255 area 0.0.0.0
network 192.168.101.16 255.255.255.252 area 0.0.0.10
network 192.168.101.24 255.255.255.252 area 0.0.0.10
network 192.168.101.28 255.255.255.252 area 0.0.0.10
area 0.0.0.10 virtual-link 192.168.102.2
area 0.0.0.10 virtual-link 192.168.102.2 authentication message-digest
area 0.0.0.10 virtual-link 192.168.102.2 message-digest-key 10 md5 juniper
!
```

End

在上面的拓扑中由于 **backbone** （**AREA 0**）配制 **MD5** 加密所以 Virtual Links 也需要配置 MD5 加密。

- 查看 Virtual Links 状态

admin@RSR04E-2> **show ospf interface**

nterface	State	Area	DR ID	BDR ID	Nbrs
fe-0/0/0.0	DR	0.0.0.0	192.168.102.2	192.168.102.1	1
lo0.0	DRother	0.0.0.0	0.0.0.0	0.0.0.0	0
vl-192.168.102.5	**PtToPt**	**0.0.0.0**	**0.0.0.0**	**0.0.0.0**	**1**
fe-0/0/1.0	DR	0.0.0.10	192.168.102.2	192.168.102.5	1
fe-0/0/2.0	DR	0.0.0.10	192.168.102.2	192.168.102.6	1

admin@RSR04E-2> **show ospf neighbor**

Address	Interface	State	ID	Pri	Dead
192.168.101.5	fe-0/0/0.0	Full	192.168.102.1	128	38
192.168.101.18	**vl-192.168.102.5**	**Full**	**192.168.102.5**	**1**	**35**
192.168.101.18	fe-0/0/1.0	Full	192.168.102.5	1	36
192.168.101.37	fe-0/0/2.0	Full	192.168.102.6	1	36

- Disable 一个邻居接口。

admin@RSR04E-2>**set interfaces fe-0/0/0.0 disable**

- 查看 **ospf** 邻居

admin@RSR04E-2> show ospf neighbor

Address	Interface	State	ID	Pri	Dead
192.168.101.18	**vl-192.168.102.5**	**Full**	**192.168.102.5**	**1**	**30**
192.168.101.18	fe-0/0/1.0	Full	192.168.102.5	1	30
192.168.101.37	fe-0/0/2.0	Full	192.168.102.6	1	31

9. Firewall Filters

简单来说 Route policy 过滤路由条目。firewall filters 用来过滤包，应用于接口。

9.1 firewall Filters Overview

可以根据应用根据 3、4 层信息来过滤，执行指定操作。

9.2 执行 Firewall Filter

RSR是通过 *Internet Processor ASIC* 来实现出功能。Internet Processor ASIC 支持处理大量的 3 、4 层信息作为匹配条件执行动作，处理包。查看Internet processor Asic：

```
user@Shiraz> show chassis hardware
Hardware inventory:
Item              Version  Part number  Serial number     Description
Chassis                    50375             M7i
Midplane          REV 03   710-002650   HF1437
Power Supply A    Rev 04   740-002497   LK22981           AC
Display           REV 04   710-001995   HF1278
Host                                    8a00000749a99a01  teknor
FEB               REV 08   710-002503   AL0781            Internet Processor II
FPC 0
  PIC 0           REV 04   750-002992   HC5418            4x F/E, 100 BASE-TX
  PIC 1           REV 03   750-002971   HE5256            4x OC-3 SONET, MM
```

9.2.1 firewall filter 命令格式

Family （inet | inet6）inet ipv4 的 firewall filter， inet6 是 IPv6 的 firewall filter

```
firewall {
    family inet {
        filter filter-name {
            term term-name {
                from {
```

```
                    match-conditions;
                }
                then {
                    actions;
                    action-modifiers;
                }
            }
            term term-name {
                from {
                    match-conditions;
                }
                then {
                    actions;
                    action-modifiers;
                }
            }
        }
    }
}
```

9.2.2 默认动作

每个 firewall filter 有一个隐藏的默认动作 diccard 丢弃。

```
term implicit-rule {
    then {
        discard;
    }
}
```

9.2.3 匹配条件

JUNOS 支持丰富的匹配条件

拒绝 telnet 会话

- 数字实现方式：

```
 [edit firewall family inet]
user@Shiraz# show
filter port-number {
    term deny-telnet {
        from {
```

```
            protocol 6;
            port 23;
        }
        then {
            reject;
        }
    }
}
```

● Well-known 协议号来识别

```
[edit firewall family inet]
user@Shiraz# show
filter example-filter-1 {
    term deny-telnet {
        from {
            protocol tcp;
            port telnet;
        }
        then {
            reject;
        }
    }
}
```

● 多条件匹配

```
[edit firewall family inet]
user@Shiraz# show
filter example-filter-2 {
    term allow-telnet {
        from {
            protocol tcp;
            port telnet;
        }
        then accept;
    }
    term allow-mail {
        from {
            protocol tcp;
            port smtp;
        }
```

```
        then accept;
    }
}
```

● 联合多个匹配条件

```
[edit firewall family inet]
user@Shiraz# show
filter example-filter-2 {
    term allow-telnet-and-mail {
        from {
            protocol tcp;
            port [ telnet smtp ];
        }
        then accept;
    }
}
```

● **Numeric Range Match Conditions**

Match Condition	Description
keyword-except	Negates a match—for example, destination-port-except ***number***.
destination-port ***number***	The TCP or UDP destination port field. You cannot specify both the port and destination-port match conditions in the same term. Normally, you use this condition in conjunction with the protocol match statement to determine which protocol is being used on the port. Some common text synonyms and their port numbers are listed here: bgp (179), bootpc (68), bootps (67), domain (53), finger (79), ftp (21), ftp-data (20), http (80), https (443), kerberos-sec (88), ldap (389), msdp (639), netbios-dgm (138), netbios-ns (137), netbios-ssn (139), nntp (119), ntp (123), pop3 (110), pptp (1723), radius (1812), rip (520), smtp (25), snmp (161), snmptrap (162), socks (1080), ssh (22), syslog (514), tacacs-ds (65), telnet (23), or tftp (69).
dscp ***number***	The Differentiated Services codepoint. The Diffserv protocol uses the type of service (ToS) byte in the IP header. The most significant six bits of this byte form the Diffserv codepoint (DSCP). In place of the numeric value, you can specify one of the following text synonyms (the field values are

Match Condition	Description
	also listed):The Expedited Forwarding RFC defines one codepoint: ef (46).The Assured Forwarding RFC defines 4 classes, with 3 drop precedences in each class, for a total of 12 codepoints: af11 (10), af12 (12), af13 (14), af21 (18), af22 (20), af23 (22), af31 (26), af32 (28), af33 (30), af41 (34), af42 (36), or af43 (38).
fragment-offset ***number***	The fragment offset field.
icmp-code ***number***	The ICMP code field. This value or keyword provides more specific information than the icmp-type condition. Because the value's meaning depends on the associated icmp-type, it must also be specified along with the icmp-code.
icmp-type ***number***	The ICMP packet type field. Normally, you specify this match in conjunction with the protocol match condition to determine which protocol is being used on the port. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): echo-reply (0), echo-request (8), info-reply (16), info-request (15), mask-request (17), mask-reply (18), parameter-problem (12), redirect (5), router-advertisement (9), router-solicit (10), source-quench (4), time-exceeded (11), timestamp (13), timestamp-reply (14), or unreachable (3).
interface-group ***group-number***	The interface group on which the packet was received. An interface-group is a set of one or more logical interfaces.
packet-length ***bytes***	The length of the received packet, in bytes. The length refers only to the IP packet, including the packet header, and does not include any Layer 2 encapsulation overhead.
port ***number***	The TCP or UDP source or destination port field. You cannot specify both the port match and either the destination-port or source-port match conditions in the same term. Normally, you specify this match in conjunction with the

Match Condition	Description
	protocol match statement to determine which protocol is being used on the port. In place of the numeric value, you can specify one of the text synonyms listed under destination-port.
precedence *ip-precedence-field*	The IP precedence field. The precedence bits are the three most significant bits in the type of service (ToS) byte in the IP header. In place of the numeric field value, you can specify one of the following text synonyms (the field values are also listed): critical-ecp (0xa0), flash (0x60), flash-override (0x80), immediate (0x40), internet-control (0xc0), net-control (0xe0), priority (0x20), or routine (0x00).
protocol ***number***	The IP protocol field. In place of the numeric value, you can specify one of the following text synonyms (the field values are also listed): egp (8), esp (50), gre (47), icmp (1), igmp (2), ipip (4), ipv6 (41), ospf (89), pim (103), rsvp (46), tcp (6), or udp (17).
source-port ***number***	The TCP or UDP source port field. You cannot specify the port and source-port match conditions in the same term. Normally, you specify this match in conjunction with the protocol match statement to determine which protocol is being used on the port. In place of the numeric field, you can specify one of the text synonyms listed under destination-port.

- 地址匹配条件

```
user@Shiraz# show
filter example-filter-3 {
    term deny-subnets {
        from {
            source-address {
                10.1.1.0/24;
                172.16.0.0/16;
            }
            destination-address {
                192.168.1.0/24;
            }
```

```
        }
        then {
            reject;
        }
    }
}
```

Match Condition	Description
address *prefix*	The IP source or destination address field. You cannot specify both the address and the destination-address or source-address match conditions in the same term.
destination-address *prefix*	The IP destination address field. You cannot specify the destination-address and address match conditions in the same term.
destination-prefix-list *prefix-list*	The IP destination prefix list field. You cannot specify the destination-prefix-list and prefix-list match conditions in the same term.
prefix-list *prefix-list*	The IP source or destination prefix list field. You cannot specify both the prefix-list and the destination-prefix-list or source-prefix-list match conditions in the same term.
source-address *prefix*	The IP source address field. You cannot specify the source-address and address match conditions in the same rule.
source-prefix-list *prefix-list*	The IP source prefix list field. You cannot specify the source-prefix-list and prefix-list match conditions in the same term.

9.2.4 动作

匹配条件后执行动作有三种：**terminating, flow-control, and action modifiers**

9.2.4.1 terminating Actions

accept, discard, and reject。三个。 Discard 和 reject 不同点 reject 是有ICMP 应答的。

9.2.4.2 Flow-Control Actions

使用 **Flow-Control Actions** next term 来控制动作。

看下面的例子：

（1）。只会记录所有的 log。

```
[edit firewall family inet]
user@Shiraz# show
filter example-filter-9{
    term log-all-packets {
        then log;
    }
    term deny-telnet {
        from {
            protocol tcp;
            port telnet;
        }
        then {
            reject;
        }
    term accept-everything-else {
        then accept;
    }
}
```

（2）记录 log 然后执行下列 term。

```
edit firewall family inet]
user@Shiraz# show
filter example-filter-9{
    term log-all-packets {
        then {
            log;
            next term;
    }
    term deny-telnet {
        from {
            protocol tcp;
            port telnet;
        }
        then {
            reject;
        }
    term accept-everything-else {
        then accept;
```

```
    }
}
```

9.2.4.2 Action Modifiers

count ： 记录原地址 10.0.0/24 的传输计数器。

```
[edit firewall family inet]
user@Shiraz# show
filter inbound-from-peer {
    term count-traffic {
        from {
            source-address {
                10.0.0/24;
            }
        then {
            count traffic-counter;
            accept;
        }
    }
}
```

Log：记录储存 memory-resident buffer that is 500 lines。通过 show firewall log 查看。

```
[edit firewall family inet]
user@Shiraz# show
filter log-tcp-flow-start {
    term count-syn {
        from {
            protocol tcp;
            tcp-initial;
        }
        then {
            log;
            accept;
        }
    }
}
```

Sample：取样。

```
[edit firewall family inet]
user@Shiraz# show
filter sample-peer-traffic {
    term peer-connections {
        from {
            source-address {
                10.10.0.0/16;
                172.30.45.0/24;
```

```
                    192.168.164.0/20;
                }
            }
            then {
                sample;
                accept;
            }
        }
}
```

Syslog：

记录 log

```
[edit firewall family inet]
user@Shiraz# show
filter log-tcp-flow-start {
    term count-syn {
        from {
            protocol tcp;
            tcp-initial;
        }
        then {
            syslog;
            accept;
        }
    }

}
```

9.2.3 Applying Firewall Filters

应该 firewall filters to interface；

```
[edit interfaces fe-0/0/0]
user@Shiraz# show
description "Connection to AS 65000";
unit 0 {
    family inet {
        filter {
            input AS65000-inbound-filter;
            output AS65000-outbound-filter;
        }
        address 10.10.10.1/24;
    }
```

```
}
```

9.3 Protecting the Routing Engine

通过过滤 lop0 来保护路由引擎。

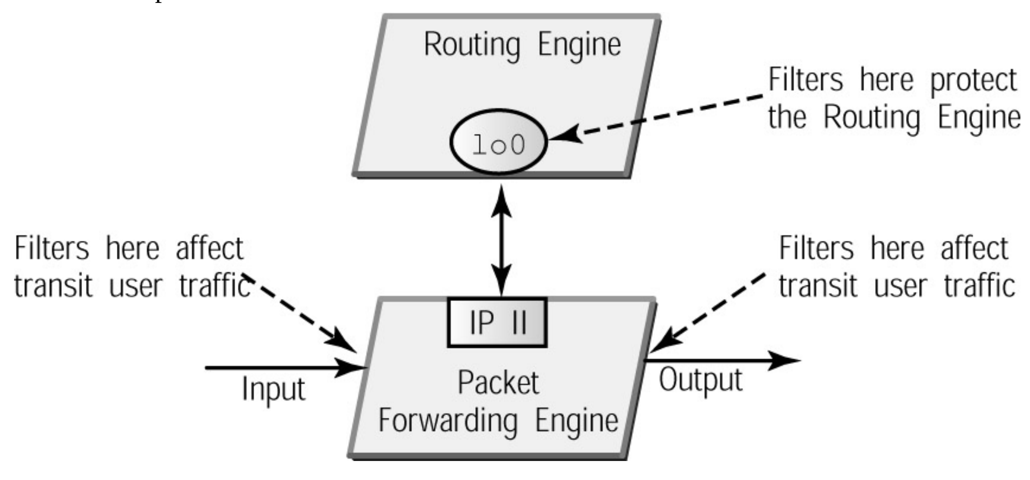


图 9-1

例子：

```
edit interfaces lo0]
user@Shiraz# show
unit 0 {
    family inet {
        filter {
            input protect-routing-engine;
        }
        address 192.168.1.1/32;
    }
}
```

9.4 Rate Limits

两个参数控制 bandwidth-limit ， burst-size-limit，控制范围 between 1500 and 100,000,000 (100MB).

一个简单的例子：

```
 [edit firewall family inet]

user@Shiraz# show
filter limit-ftp {
```

```
    policer policer-1 {
        if-exceeding {
            bandwidth-limit 400k;   /**正常使用带宽 **/
            burst-size-limit 100k; /**超出正常带宽后带宽 **/
        }
        then discard;    /*** 如果应用超过 500k 后丢弃  ***/
    }
    term ftp {
        from {
            source-address {
                10.2.3/24;
            }
            protocol tcp;
            destination-port [ftp ftp-data];
        }
        then {
            policer policer-1;  /*** 应用 FTP **/
            accept;
        }
    }
    term accept-all {
        then accept;
    }
}
```

直接应用到 interface

```
[edit firewall]
user@Shiraz# show
policer police-all-traffic {
    if-exceeding {
        bandwidth-limit 10m;
        burst-size-limit 100k;
    }
    then {
        discard;
    }
}
[edit interfaces fe-0/0/0]
user@Shiraz# show
description "Connection to Customer-A";
unit 0 {
    family inet {
        policer {
            input police-all-traffic;
```

```
            output police-all-traffic;
        }
        address 10.100.1.1/24;
    }
}
```

9.5 show firewall

```
user@Shiraz> show firewall
Filter/Counter              Packet count              Byte count
protect-re
  deny                                46                    2791
  icmp                                 0                      NA
```

9.5.1 show firewall log

```
user@Shiraz> show firewall log
Time      Filter A Interface        Pro Source address  Destination address
11:56:47 pfe    A so-0/1/0.0        OSP 200.0.0.2        224.0.0.5
11:56:43 pfe    A so-0/1/1.0        OSP 200.0.0.1        224.0.0.5
11:56:42 pfe    A fe-0/0/1.0        TCP 10.0.8.2         10.0.8.1:179

user@Shiraz> show firewall log detail
2001-09-03 11:57:42 UTC, Filter: pfe, Action: accept
    Interface: so-0/1/1.0, Protocol: OSPF, Length: 64
    Source: 200.0.0.1, Destination: 224.0.0.5
2001-09-03 11:57:42 UTC, Filter: pfe, Action: accept
    Interface: fe-0/0/1.0, Protocol: TCP, Length: 52
    Source: 10.0.8.2:1024, Destination: 10.0.8.1:179
```

9.5.2 show log messages

```
user@Shiraz> show log messages | match FW
Sep  2 16:38:39 router scb FW: fe-0/0/2.0   A  tcp 10.0.2.1 192.168.5.1  1026
23 (2 packets)
Sep  2 16:38:40 router scb FW: fe-0/0/2.0   A ospf 10.0.2.1 224.0.0.5      0
0 (1 packets)
Sep  2 16:38:40 router scb FW: fe-0/0/2.0   A  tcp 10.0.2.1 192.168.5.1  1026
```

```
23 (2 packets)
```

9.5.3 show interfaces filters

```
user@Shiraz> show interfaces filters
Interface        Admin Link Proto Input Filter          Output Filter
fe-0/0/0         up    up
fe-0/0/0.0       up    up    inet  filter-1
   filter-2
fe-0/0/1         up    up
fe-0/0/1.0       up    up    inet  filter-3
   filter-4
fe-0/0/2         up    down
fe-0/0/3         up    down

user@Shiraz> show interfaces policers
Interface        Admin Link Proto Input Policer          Output Policer
fe-0/0/0         up    up
fe-0/0/0.0       up    up    inet  fe-0/0/0.0-in-policer
   fe-0/0/0.0-out-policer
fe-0/0/1         up    up
fe-0/0/1.0       up    up    inet  fe-0/0/1.0-in-policer
   fe-0/0/1.0-out-policer

fe-0/0/2         up    down
fe-0/0/3         up    down
user@Shiraz> show policer
Policer: so-2/2/0.0-in-policer
    so-2/2/0.0-in-policer
                    0 packets
Policer: so-2/2/0.0-out-policer
    so-2/2/0.0-out-policer
                    238 packets
```

9.6 Configuring Filter-Based Forwarding

策略路由。通过下面是配置例子介绍。

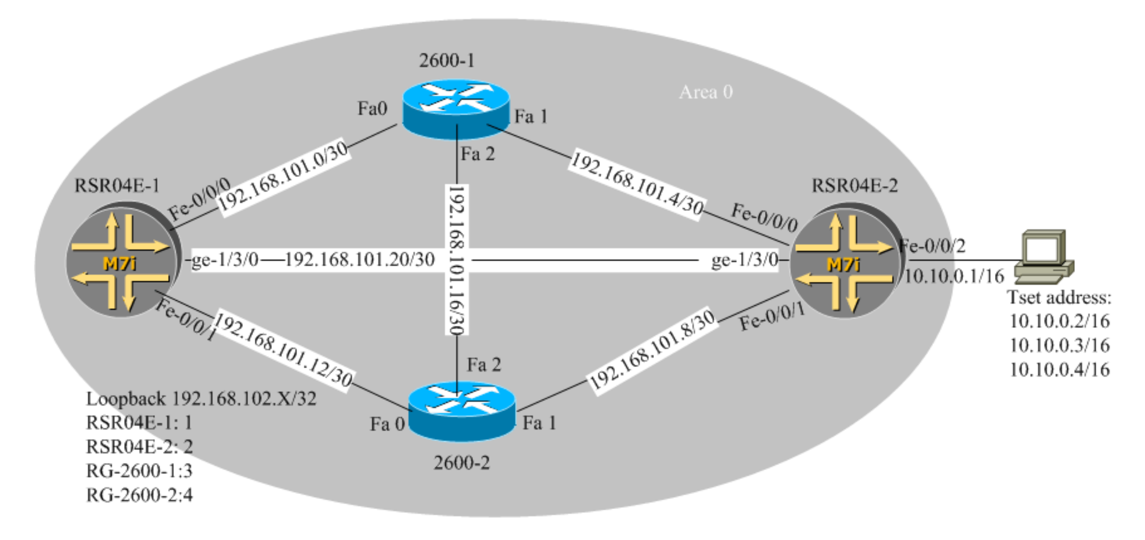


图 9－2

需求如下：

在 RSR04E-2 实现

1，如果源地址是 10.10.0.2 这个地址 去 RSR04E-1 的 lookback 测试地址 193.1.1.1，RSR04E-2 把包发到 2600-2 接口 192.168.101.9
2，如果源地址是 10.10.0.3 这个地址 去 RSR04E-1 的 lookback 测试地址 193.1.1.1，RSR04E-2 把包发到 2600-1 接口 192.168.101.5
3，其它的源地址根据动态路由走。也就是把包发送到 RSR04E-1 ge-1/3/0 地址 192.168.101.20。

实现这些功能需要在 RSR04 配置下列步骤：

1，配置 routing-instances

```
routing-instances {
    route-source-filter1 {       /*** 名字   ***/
        instance-type forwarding;     /*** 类型转发 ***/
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 192.168.101.9;
            }
        }
    }
    route-source-filter2 {              /*** 名字   ***/
        instance-type forwarding;    /*** 类型为转发 ***/
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 192.168.101.5;
            }
        }
    }
}
```

导入接口路由表

```
routing-options {
```

```
    interface-routes {
        rib-group inet source-filter;
    }
    rib-groups {
        source-filter {
            import-rib [ route-source-filter1.inet.0
            route-source-filter2.inet.0 inet.0 ];
        }
    }
 }
```

2,定义过滤器;

```
firewall {
    family inet {
        filter classify-source {
            term isp1 {
                from {
                    source-address {
                        10.10.0.2/32;
                    }
                }
                then routing-instance route-source-filter1;
            }
            term isp2 {
                from {
                    source-address {
                        10.10.0.3/32;
                    }
                }
                then routing-instance route-source-filter2;
            }
            term default {
                then accept;
            }
        }
    }
}
```

4，应用到具体接口;

```
interfaces {
    fe-0/0/2 {
        unit 0 {
            description to-3550-fa0/48;
            family inet {
                filter {
                    input classify-source;
```

```
                }
                address 10.10.0.1/16;
            }
        }
    }
}
```

5，使用 traceroute 测试：

5.1 以源地址 10.10.0.2 traceroute 结果：

```
RG-S3550#traceroute 193.1.1.1
    1    12ms    1ms    1ms    10.10.0.1
    2    1ms     4ms    1ms    192.168.101.9
    3    1ms     2ms    1ms    193.1.1.1
Trace complete successfully.
```

5.2 以源地址 10.10.0.3 traceroute 结果：

```
RG-S3550#traceroute 193.1.1.1
    1    29ms    1ms    1ms    10.10.0.1
    2    1ms     1ms    1ms    192.168.101.5
    3    2ms     1ms    4ms    193.1.1.1
Trace complete successfully.
```

5.3 以源地址 10.10.0.3 traceroute 结果：

```
RG-S3550#traceroute 193.1.1.1
    1    3ms     1ms    1ms    10.10.0.1
    2    1ms     1ms    3ms    193.1.1.1
Trace complete successfully.
```

RSR04E-2 完整配置：

```
admin@RSR04E-2# show
version 7.3B1.1;
system {
    host-name RSR04E-2;
    root-authentication {
        encrypted-password "$1$4lSSr3rQ$dwkAWDsD6HZnsgdH7l.Zv1"; ##
SECRET-DATA
    }
    login {
        user admin {
            uid 2004;
            class super-user;
```

```
            authentication {
                encrypted-password "$1$sGeRZFVP$Gpey23qE6imRCBXFInu3f/";
## SECRET-DATA
            }
        }
    }
    services {
        ftp {
            connection-limit 10;
            rate-limit 10;
        }
        ssh {
            protocol-version v2;
        }
        telnet;
    }
}
interfaces {
    fe-0/0/0 {
        description to-2600-1-fa1;
        unit 0 {
            family inet {
                address 192.168.101.6/30;
            }
        }
    }
    fe-0/0/1 {
        unit 0 {
            description to-2600-2-fa1;
            family inet {
                address 192.168.101.10/30;
            }
        }
    }
    fe-0/0/2 {
        unit 0 {
            description to-3550-fa0/48;
            family inet {
                filter {
                    input classify-source;
                }
                address 10.10.0.1/16;
            }
        }
```

```
    }
    ge-1/3/0 {
        description to-RSR04E-1-ge-1/3/0;
        unit 0 {
            family inet {
                address 192.168.101.22/30;
            }
        }
    }
    fxp0 {
        description IMG;
        unit 0 {
            family inet {
                address 192.168.0.12/24;
            }
        }
    }
    lo0 {
        description RID;
        unit 0 {
            family inet {
                address 192.168.102.2/32;
            }
        }
    }
}
routing-options {
    interface-routes {
        rib-group inet source-filter;
    }
    rib-groups {
        source-filter {
            import-rib [ route-source-filter1.inet.0
route-source-filter2.inet.0 inet.0 ];
        }
    }
    router-id 192.168.102.2;
}
protocols {
    ospf {
        area 0.0.0.0 {
            interface fe-0/0/0.0;
            interface fe-0/0/1.0;
            interface lo0.0 {
```

```
                passive;
            }
            interface fe-0/0/2.0 {
                passive;
            }
            interface ge-1/3/0.0;
        }
    }
}
firewall {
    family inet {
        filter classify-source {
            term isp1 {
                from {
                    source-address {
                        10.10.0.2/32;
                    }
                }
                then routing-instance route-source-filter1;
            }
            term isp2 {
                from {
                    source-address {
                        10.10.0.3/32;
                    }
                }
                then routing-instance route-source-filter2;
            }
            term default {
                then accept;
            }
        }
    }
}
routing-instances {
    route-source-filter1 {
        instance-type forwarding;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 192.168.101.9;
            }
        }
    }
    route-source-filter2 {
```

```
        instance-type forwarding;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 192.168.101.5;
            }
        }
    }
}

[edit]
```

如果实现基于目标地址的策略路由通过适当的修改上面的某些参数；就能实现。